人脸识别与人体动作识别技术及应用

曹　林　著

電子工業出版社
Publishing House of Electronics Industry
北京・BEIJING

内 容 简 介

本书以模式识别的一些基本理论与方法为基础，重点讨论了模式识别在人脸识别、人脸配准、人脸检测、素描人脸识别、图像超分辨率重建、Kinect 人体动作识别中的应用。全书共分 7 章，第 1 章概述了人脸识别技术与人脸图像超分辨率重建技术的发展现状。第 2 章提出了基于人脸纹理特征点 3D 人脸配准算法和基于均值权重粒子滤波器的人脸检测跟踪算法。第 3 章提出了基于 LBP 的素描人脸识别算法。第 4 章提出了一种基于 Gabor 小波变换和隐马尔可夫模型的人脸识别算法。第 5 章提出了一种基于分块 PCA 的单帧人脸超分辨率算法。第 6 章提出了基于空间几何角度信息的人体动作识别算法。第 7 章实现了一种基于 Kinect 的手势识别算法，完成了对智能小车的控制。

本书可供从事图像理解与识别、生物特征识别及相关研究和开发的教师、研究生及工程技术人员参考。

图书在版编目（CIP）数据

人脸识别与人体动作识别技术及应用 / 曹林著. —北京：电子工业出版社，2015.8

ISBN 978-7-121-26660-7

Ⅰ. ①人… Ⅱ. ①曹… Ⅲ. ①面—图像识别 ②人体—运动—图像识别 Ⅳ. ①TP391.41

中国版本图书馆 CIP 数据核字（2015）第 161226 号

策划编辑：董亚峰
责任编辑：郝黎明
印　　刷：
装　　订：
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编 100036
开　　本：720×1 000　1/16　印张：13.25　字数：339.2 千字
版　　次：2015 年 8 月第 1 版
印　　次：2015 年 8 月第 1 次印刷
定　　价：48.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

展　望

本书以模式识别的一些基本理论与方法为基础，重点讨论了模式识别在图像配准、人脸检测、素描人脸识别、图像超分辨率重建、Kinect 人体动作识别中的应用。其中两个章节涉及小波分析领域关于人脸识别的应用。现如今人脸识别及人体行为动作研究已经取得了一定研究成果，但是这并不说明它的弊端问题已经得以解决。

以物体之间的遮挡为例，此时人体部分被遮挡，如何剥离与之重叠的物体，并完整地分割出人体目标将是一个挑战。可以将监控视频中目标人体各个部位的定位与分割作为研究的切入点，通过对大量人体样本的训练与学习，从而精准地识别出人体的各个部位，并由此构建出人体骨架。对于实时性来讲，它受多方面因素的影响，如硬件条件、算法的复杂度、算法的识别效率，其间需要找到一个平衡点来处理这些影响。另外，图像配准在实际应用中的严格性也是一个挑战，由于它的准确度很多情况下并不尽如人意，因此，图像配准常常需要手工矫正，自动图像配准技术仍然是一个难题。

问题的存在其实并不能掩盖它们在实用性方面的价值，人脸识别技术在生物特征识别领域、司法领域、访问权限控制领域等均有应用。很显然，传统的人脸识别已经趋近成熟，这势必带来研究热点的一个新的转移。目前，异质人脸识别领域的素描人脸识别以及漫画人脸识别正在悄然兴起，这无疑给人脸识别研究注入新的动力。然而，异质人脸识别研究并不像传统研究那样简单，作为新兴事物，它正面临着一系列的挑战。用于刑侦的素描人脸识别与常规人脸识别不同，素描人脸识别通常都是正向的，姿态影响较小；表情通常为中性表情，表情变化较小；光照条件人为可控，基本不受光照影响。素描人脸识别的困难在于，刑侦专家根据目击者的描述所画的图像与真实图像之间的“像几分”问题。现场临摹的素描图与真实人像之间的相似度较大，相对识别效果较好；事后回忆的素描图与真实人像之间的差异偏大，识别率

必然降低，但实际应用中，刑侦素描人脸识别往往不需要 Rank-1，因此识别率可通过 Rank-N 提高。

未来，合成素描照，即通过软件处理或其他手段得到的素描照，或许会提高法医素描照识别的可操作性。它可以通过 CBR（Component Based Representation）来处理原图和合成素描图之间的形态差距，以 ASM 检测人脸的局部信息，以 MLBP 捕捉纹理和结构特征作为切入点，这里不同局部图像之间存在的相关性将会是未来研究的重点。但是年龄问题仍然是亟待解决和不可避免的，同一个人不同时间段的照片（与素描照）匹配度能否理想也是未来研究的一个方向。

当前，基于原型的可操作异质人脸识别框架表明，异质面部特征或许可以用到异质人脸图像匹配上。当在复杂场景下进行红外热像与原图匹配时，就需要考虑到诸多不确定因素，这仍然是目前研究努力的一个方向，就像深度图像的人脸匹配需要使用到 LIDAR 传感器。已经证明这种传感器在远距离捕获高分辨率图像上具有较为优秀的能力，这将在人工智能及法律实施应用上形成一种可实施的人脸识别场景。

异质人脸识别领域的另一热点——漫画人脸识别技术或许会成为未来研究的一个新方向，随着大量数据库的开放，将有助于一些研究者找到新的算法来处理这类问题。对于漫画人脸识别，单一的算法是很难做到较高的识别率的，不同算法的结合应该会是新的思路。也许我们可以利用漫画人脸与原图特征点的相对位置，及各特征点之间的相对距离，探索新的研究方式。

另外，基于 Kinect 的人体动作研究目前大都针对于单个人体动作的识别，研究层面尚且较浅。很显然，实时的实现多目标的人体动作识别或将成为目标检测新的研究热点。例如，小组人群目标的检测，即多个人在同一时刻表现出不同的行为，若将其实时的检测出来，而不是分别对单个目标检测后组合，这样检测识别效率明显提高，应用性势必会得到提升。就社会意义来讲，人体动作识别的最终目标即是实现异常目标人体动作的识别，然而这仍然任重道远。

图像超分辨率重建技术目前处于研究初级阶段，由于其社会意义重大，很多研究者们仍然希望找到更为高速高效的算法，如盲超分辨率重建方法、概率模型方法等。实际情况下，人们更需要在“盲的”情况下进行超分辨率重建，对于重建质量，仍然需要进一步提高。未来继续在运动估计、退化模型、压缩视频、算法优化等方面进行更深入的研究或许会取得新的进展。

前　言

人脸识别与人体动作因其在公安、视频监控、人机交互等方面良好的应用前景而成为近年来模式识别、图像处理等领域中的研究热点。虽然人类可以毫无困难地通过人脸而识别出某个人，但要建立一个能够完全自动进行人脸识别的人工智能系统却非常困难。困难的原因在于人类对视觉认知机理的了解还很肤浅，还不知道如何用数学来准确描述认知现象。使用图像处理技术来进行人脸识别时，困难表现在：人脸光照模式的不确定性，人脸表情的多样性和人脸姿态的随意性。到目前为止，已经取得的研究成果离这一问题的彻底解决还有很大的距离。

本书以模式识别的一些基本理论与方法为基础，重点讨论了模式识别在图像配准、人脸检测、素描人脸识别、图像超分辨率重建、Kinect 人体动作识别中的应用。全书共分 7 章，各章节主要内容如下：

第 1 章概述了人脸识别技术与人脸图像超分辨率重建技术的发展现状，并介绍了人体行为识别的研究进展。

第 2 章提出了基于人脸纹理特征点 3D 人脸配准算法。通过对 3D 人脸投射到 XOY 平面，获取正面人脸纹理图，利用 VOSM 算法实现了 3D 人脸数据库的重建以及配准。提出基于均值权重粒子滤波器的人脸检测跟踪算法，提高了系统对于人脸跟踪的准确度。

第 3 章提出了一种新的素描人脸识别，以 LBP 算子描述素描人脸和光学照人脸的相似性，采用 DOG 滤波器进一步提升了图像纹理效果。利用机器学习思想，加入分块特征，实现了素描人脸识别。提出了基于 SIFT 特征的人脸验证算法，以 SIFT 特征为基础，划分为数量特征和位置特征进行验证，实现了人脸验证。

第 4 章提出了一种基于 Gabor 小波变换和隐马尔可夫模型的人脸识别算法，以及基于 Gabor 小波变换、独立元分析和隐马尔可夫模型的人脸识别方

法。上述方法识别率高，复杂度较低，对部分遮挡的图像具有较大的容忍度。

第 5 章提出了一种基于分块 PCA 的单帧人脸超分辨率算法。此算法用人脸图像块位置信息表征人脸图像的全局结构，图像块内容表征人脸图像的细节，进而重建出高分辨率人脸图像。

第 6 章提出了一种基于深度传感器提取人体骨骼空间角度信息的动作识别方法和一种基于三维时空特征的人体行为识别算法。上述算法着眼于虚拟现实、人机交互中的人体动作识别，利用空间几何角度来实现人体动作识别。

第 7 章实现了一种基于 Kinect 的手势识别算法，完成了对智能小车的控制。

本书的出版得到了北京市属高校青年拔尖人才培育计划（项目号：CIT&TCD201304119）、国家科技重大专项煤层气田地面集输信息集成及深度开发技术（项目编号：2011ZX05039-004-02）、人才培养项目—学科与研究生教育水平提高项目（项目号：5111524100）等科研项目的资助，在此一并表示感谢。

由于时间仓促，书中难免存在不足，欢迎读者对本书批评指正。

目 录

第 1 章

绪　论

1.1　人脸识别技术的研究与应用

现代社会中，随着计算机技术的高速发展，计算机技术已经融入了人们生活的方方面面，人们和计算机进行交互的场合也越来越多，人工智能技术成为了许多学者研究的热点和重点。让计算机像人类一样思考，是人们一直以来在努力实现的目标，从“深蓝”象棋机器人的诞生，到现在的人工智能机器人可以和人进行简单交流，人工智能技术已经取得了长足的发展。同时，随着人工智能的发展，利用计算机代替人工作，也是人们对人工智能技术发展的一个重要需求。安全问题是各个国家都十分看重的问题，而人工智能在安全问题方面可以提供非常大的帮助，智能安防系统在近年来已经进入人们的生活中。其中，人脸识别[1]技术，正是人工智能领域的关键技术。人脸识别的相关技术包括人脸检测和跟踪，人脸验证以及各类人脸识别等，这些技术都可以广泛地应用于智能机器人、智能视频监控系统、门禁系统中。因此人脸识别不仅是一项拥有极高学术研究价值的技术，还是一个有巨大商业价值的技术。尽管人脸识别技术已经发展多年，但是还未能达到人们预期的目标。由于以上原因，人们对于人脸识别技术研究从未停止。本书在前人的研究基础上，对人脸识别技术进行了深入的研究，为了全面地研究人脸识别技术，实现人脸识别的整个流程，下面按图像配准、人脸检测和跟踪、人脸验证以及人脸识别的顺序，完成人脸识别技术所有步骤，对人脸识别技术的各个方面进行了完整的展现。

1.1.1 国内外人脸库介绍

人脸识别技术属于大数据技术的一种，因此作为人脸识别的基础，人脸库的建立不可缺少。国内外已经建立许多权威人脸库，具体介绍如下：

（1）FERET 人脸数据库。本数据库是由美国军方建立的人脸数据库，共有14000 多幅图片，每人对应 7 幅图像，在不同的姿态、表情和光照条件下采集。采集的人脸多数是由西方人构成的，人种较为单一。

（2）CMU-PIE 人脸数据库。本数据库由美国卡内基梅隆大学所建立，库中共有 68 名志愿者共 41368 幅图像，每幅图像的姿态和光照条件都进行了严格控制。

（3）YALE 人脸数据库。由耶鲁大学建立的人脸数据库，包含了 15 名志愿者以及对应的 165 幅图像。

（4）ORL 人脸数据库。由剑桥大学的 AT&T 实验室所建立的人脸数据库，由 40 名志愿者以及对应的 400 幅图像所组成，是人脸识别研究最常用的人脸数据库之一。

（5）MIT 人脸数据库。由麻省理工大学所建立的人脸数据库，包括 16 名志愿者以及对应的 2592 幅不同姿态、光照等条件的图像。

（6）CAS-PEAL 人脸数据库。中国最大的人脸数据库[2]，是中国人脸库中最具权威的数据库。该数据库由中科院计算所建立，包括 1040 名志愿者，对应不同表情、光照、年龄、饰物等条件的共 99450 幅图像。

（7）BJUT-3D 3D 人脸数据库。由北京工业大学建立的第一个中国人的 3D 人脸数据库[3]，是国内常用 3D 人脸数据库，有男女各 250 名，共 500 幅 3D 人脸图像。

（8）CHUK 素描人脸数据库。由香港大学所建立的素描人脸数据库[4]，主要用于研究素描人脸识别，数据库包含有 606 人，分别来自香港大学人脸数据库以及其他数据库，每人都有一幅常规光学人脸图像以及对应的一幅素描人脸图像。

还有其他一些常用数据库，因为篇幅有限，没有一一列举。

1.1.2 国内外研究现状

早在 1964 年，国外就开始了对人脸识别的相关研究。研究初级人脸识别并没有单独作为一个研究领域，只是作为一般性的模式识别问题进行研究，方法也大多是针对人脸几何特征实现的算法。进入 20 世纪 90 年代后，人脸研究突飞猛进，不仅建立了数个大型人脸库的建立，而且出现了一些商业化的人脸识别系统。

麻省理工大学在这个阶段提出的特征人脸算法（Eigenface）被公认为是经典的人脸识别算法。与此同时，由 Belhumeur 等提出的 PCA 结合 LDA 的人脸识别算法，也是人脸识别算法的一个里程碑[5, 6]。在 20 世纪 90 年代末，Viola 提出了 AdaBoost 算法[7]，在人脸检测方面取得了突破性进展，为后续人脸检测的相关研究奠定了基础。除此之外，还有像 Gabor、神经元网络等方法。20 世纪后，3D 人脸识别的相关研究开始崭露头角，3D 人脸识别、3D 人脸检测等算法也相应出现。在 3D 人脸研究方面做出突出贡献的 Blanz 和 Vetter ，最先提出了基于 3D 变形[8, 9]（3D Morphable Model）模型人脸识别，解决了人脸检测与识别难以解决的多姿态问题。同时，在这一阶段，更多的 2D 人脸识别算法也相应提出，支持向量机（Support Vector Machine，SVM）[10]作为统计学习理论的代表算法，是最为突出的。

相比于国外，国内的发展相对较晚，但是发展迅速，在 20 世纪 90 年代起，国内开始在人脸技术方面进行相关研究，各个研究机构开始建立。最有代表性的研究机构以及学校包括，清华大学计算机系、自动化系和电子系，代表人物有徐光佑教授、边肇祺教授等；哈尔滨工业大学计算机系和中科院计算技术研究所合作的研究组，代表人物有高文教授、陈熙霖教授等；中科院自动化研究所模式识别国家重点实验室，代表人物有谭铁牛博士、王蕴红博士等；北京交通大学，代表人物有袁保宗教授等；北京工业大学，代表人物有沈兰荪教授、尹宝才教授等。还有许多研究机构和学校就不一一列举，这些研究机构，研究小组和学校在人脸识别以及其他模式识别领域进行了许多有意义的尝试，取得了非常多的研究成果。

1.1.3　人脸识别技术的难点和发展趋势

尽管人脸识别以及相关的研究已经取得了非常多的成果，但是仍有一些研究难点并未解决。

1．光照影响

光照因素对于各类人脸研究的影响是非常巨大的，无论是检测、识别还是验证。在光照条件不理想（过强或者过弱）的情况输入待处理图像，会导致结果误差增大。目前解决这一问题的方法有两种：第一是通过算法对图像进行光照补偿处理，消除部分光照影响；第二是通过外部条件实现，如让检测、识别以及验证人脸区域的光照保持在合理程度，或者直接采用不受光照影响的红外图像等来实现算法。在这两种方法中，第一种节约成本但是效果不好，第二种

效果更佳，但是成本更大。

2．姿态影响

人脸的姿态会影响到各类人脸识别算法的效果，因为人脸识别的研究基础是人脸数据库，而人脸数据库中收集的人脸一般都是正面姿态的人脸，所以对其他姿态的人脸容忍度都比较低。针对这个问题，一般在训练的时候，加入其他类型姿态的人脸，增加算法对多姿态人脸的鲁棒性。另一类方法，采用 3D 人脸可以有效地解决多姿态问题，因为 3D 人脸本身具有可旋转性，可改变模型的姿态，但是 3D 人脸获取困难，很难在一般设备上得到普及。

3．目标图像的分辨率影响

一般来说，目标图像的分辨率越高，目标人脸保留的信息也越多，因此人脸识别率更高。相反的分辨率低的图像会造成识别率下降。增强监控设备获取图像的像素是一个很好的解决手段，然而在一些情况下获取的图像往往是低分辨率的，因此针对这个情况，一些学者提出了图像超分辨重建算法，通过训练的方法由低分辨率的图像获取高分辨率的图像。

与此同时，人脸识别的发展也取得了长足的进步，主要从深度和广度两个方面分析：

（1）深度方面，人脸识别的相关技术已经和人们生活息息相关。人脸识别的相关技术可用于安防系统，譬如门禁系统、监控系统等。因此人脸识别相关技术已经走向了商业化、产业化，同时算法的稳定性，兼容性也在不断提高。

（2）广度方面，人脸识别技术也从开始的 2D 人脸识别、同质人脸识别等，发展到 3D 人脸识别以及异质人脸识别。3D 人脸识别技术在虚拟现实领域发挥巨大的作用，同时异质人脸识别的相关技术也能为医学、刑侦等领域提供巨大的帮助。因此人脸识别的相关技术将在未来扮演更加重要的角色。

1.2　人脸图像超分辨率重建技术的研究与实现

数字图像处理是一门迅速发展的学科，它与人类的生活越来越密不可分，随着对计算机研究的不断深入，图像处理已经在很多领域中广泛应用，如机器视觉、工业检测、遥感卫星，图像传输、医学图像及视频监控等。

图像是经过图像观测系统，以不同形式和手段对真实的客观世界进行二维平面投影得到的。现实生活中，很多因素会不可避免地造成图像质量的下降，这些因素包括传感器的形状和尺寸、光照的影响、运动模糊、噪声干扰等，因此获得一幅高分辨率的图像几乎是不可能的。提高图像分辨率最直接的方法是提高成像装置的分辨率，但是受传感器阵列排列密度的限制，提高传感器的空间分辨率越来越难，通常采用的方法是减少单位像素的尺寸（即增加单位面积内的像素数量），例如：对于数字摄像机来说，就是减少其传感单元的尺寸，从而提高传感器的阵列密度，使其能够分辨出更多场景细节，但是成本会很高。另外，现有的技术工艺也限制了图像分辨率的进一步提高。因此从硬件方面来提高图像的分辨率是不切实际的。因此从软件方面着手来提高图像的分辨率，即超分辨率重建技术（Super Resolution Reconstruction，SRR）有着极大的现实意义和应用价值。

图像分辨率是指图像在单位长度上具有的像素数，单位一般为“像素/英寸”（pixel/inch），通常将分辨率作为衡量图像细节丰富程度的指标。图像质量的优劣和图像分辨率有着紧密的联系，图像的分辨率越高代表图像质量越好，越清晰，并且含有的高频信息越丰富。高分辨率图像的像素密度高，图像细节信息丰富，在实际应用中这些细节信息是很重要的。例如，高分辨率的医学影像可以帮助医生做出更精确的诊断；高分辨率的卫星图像有助于判断出地面上的相似建筑物或物体；高分辨率的人脸图像能够更准确有效地检测和识别人脸，在刑侦领域帮助警察更快地识别犯罪嫌疑人的相貌，加速破案。因此，在实际生活中高分辨率图像有很高的应用价值，对高分辨率图像的强烈需求有力地推动了图像超分辨率研究领域的发展[11]。

图像超分辨率重建技术就是利用软件的方法，将一幅或多幅低分辨率（Low Resolution，LR）图像重建成一幅高分辨率（High Resolution，HR）图像的过程。图 1.1 是两组低分辨率图像进行超分辨率重建后的效果图，左边的图像是低分辨率的，右边的图像的是重建的图像，通过对比可知，重建后的图像更清晰，噪声也降低了，如字母，边界部分都清晰可见。

超分辨率问题的解决涉及许多图像处理、计算机视觉、优化理论等领域中的基本问题, 例如图像配准、图像分割、图像压缩、图像特征提取、图像质量评价、机器学习、最优化算法等，超分辨率是这些基本问题的一个具体应用领域, 同时也对它们的研究进展起到了推动的作用。因此超分辨率问题本身的研究具有重要的理论意义。

图像超分辨率重建技术在很多领域都有着广阔的应用前景[11,12], 包括以下几个方面。

（a）　　（b）

图 1.1　图像超分辨率重建示意图

（1）公共安全领域。在公共安全领域（如银行、海关、机场等），视频监控技术越来越普及，在视频监控中，为了方便获取全景信息，摄像头通常离拍摄物距离较远，导致拍摄的视频图像较模糊，图像细节的信息较少，如犯罪嫌疑人的脸或肇事车辆的车牌号，这样不能快速地帮助警察侦破案件，因此，图像超分辨率技术可以应用到视频监控中，对视频图像中的关键部分如人脸或牌照进行高分辨率图像重建，为案件的侦破和事件的处理迅速提供重要的线索及证据。

（2）医学图像领域。通过各种医学成像技术［如 X 光、核磁共振（MRI）、断层图像重构（CT）和超声波等］形成的医学图像能够帮助医生诊断病情[11]，高分辨率医学图像能更精确地显示病变部位的具体位置和详细情况，能帮助医生更精确地诊断病因，制定更好的治疗方案。但受到成像设备本身的制约和成像技术的不完善，导致获取高分辨率的医学图像几乎是不可能的，因此图像超分辨率技术能为高清晰医学图像的产生提供了更有效的途径。

（3）卫星遥感图像领域。在卫星遥感图像应用中，高分辨率的图像有利于正确有效地跟踪识别目标。例如，“火星之脸”被证明只是火星上的奇怪石头；火星勘测轨道器的高分辨率成像科学实验摄像仪提供了大量清晰的火星表面图像，为科学家们对火星的深入研究提供了第一手资料。

（4）军事应用领域。虽然红外热像在现代战争中的夜间侦查监视中获得广泛的应用。但受到红外热像系统硬件的限制，获得的侦查图像的分辨率通常较低，图像模糊，不利于清晰地发现敌人，故图像超分辨率重建技术在现代化的战争中也有很广泛的应用前景。

（5）数字电视领域。现实生活中，随着数字电视的不断的普及，人们对高清晰度电视（HDTV）的需求也日益强烈。视频标准转换应用中，超分辨率技术会

进一步减少成本，享受高分辨率画面质量将会使人们身心愉悦，更好地满足人们日益增长的文化需求。

（6）历史、人文照片的复原。一些历史文献或旧照片等常常由于保存不当或人为损毁导致图像质量降低，图像超分辨率技术就可以增强图像的视觉效果，对旧照片进行恢复。

人脸是图像的一种特殊类型，具有高度相似性，人脸图像处理是数字图像处理中很典型的问题。近年来在公共安全领域，视频监控技术的应用越来越普及，但是受到监控设备自身分辨率的影响，拍摄环境的干扰和拍摄物离摄像头较远，获得的人脸图像的分辨率通常比较低，不能很好地进行人脸检测和识别，因此如何将超分辨率技术运动到人脸图像中，专门针对人脸图像的超分辨率重建技术成为新的研究热点。人脸超分辨率重建是专门针对人脸图像，能大大提高人脸图像分辨率，恢复脸部细节特征。使用人脸超分辨率技术重建后的图像去进行人脸身份确认可以得到更好的效果。另外，在人脸表情分析方面，也可以先对其进行超分辨率处理然后进行表情分析。

人脸超分辨率重建技术在许多领域有巨大的发展前景，如①证件识别，如身份证、驾驶证、护照等证件上人脸图像重建；②刑侦破案，可以提高视频图像的中嫌疑犯的人脸质量，帮助公安机关快速锁定嫌疑人；③海关、机场等对公共安全比较敏感的部门的监控系统；④高清视频会议，突出参会人员的脸部信息，增强视频会议的真实直观的效果；（⑤人脸表情分析等，人脸图像超分辨率重建技术成为图像超分辨率重建技术中一个很重要的分支，是模式识别领域一个研究热点[12]。人脸图像超分辨率重建的示意图如图 1.2 所示，图 1.2（a）是低分辨率人脸图像；图 1.2（b）是超分辨率重建后的人脸图像；图 1.2（c）是原始的高分辨率人脸图像。

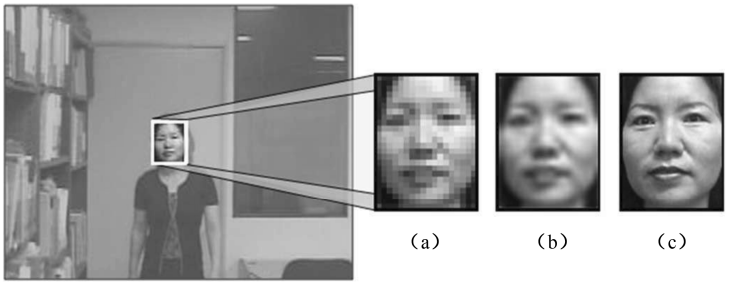


图 1.2　人脸图像超分辨率重建示意图

1.2.1 图像超分辨率的发展及国内外研究现状

图像超分辨率技术是在 20 世纪 60 年代提出的，目前已有的超分辨率重建算法主要分为基于插值的超分辨率重建算法、基于重建的超分辨率重建算法和基于学习的超分辨率重建算法三类算法，下面对这三类算法的国内外发展现状进行阐述。

基于插值的超分辨率重建算法是研究最早的算法，在实际生活中已经被广泛应用，主要用于单帧图像的重建，其原理是利用邻近像素点的灰度值来产生待插值像素点的灰度值。传统的插值算法包括最近邻域插值、双线性插值、双三次插值，这些都是建立在图像的连续性假设之上的，对满足灰度值连续性假设的区域进行插值，可以使生成的像素灰度值延续这种连续性的变化，但是对那些具有不连续灰度特性的像素进行插值时，会模糊图像的边缘和纹理，降低图像的质量[13]。为了克服传统方法的不足，许多新的插值算法被提出。Allebach 等 [14]提出了 EDI 方法（Edge-directed Interpolation，边缘导向插值）是最早出现的边缘自适应插值算法。后来 Li 和 Orchard[15] 提出了一种新的边缘导向插值算法（New Edge-directed Interpolation，NEDI），这种算法利用低分辨率图像与高分辨率图像之间在像素点协方差上的几何相关性来解决超分辨率上的问题边缘[12]。新提出的插值方法能够得到更好的图像边缘，从而提高图像的视觉效果。总之，插值算法简单、效率高、实时性好，但并没有考虑到图像的成像过程，重建图像的效果不太理想。

基于重建的超分辨率重建算法是目前研究最多的算法，主要结合图像的先验知识进行重建。Tsai 和 Huang[17]首次提出了序列图像超分辨率重建的方法[12]，采用频域解混叠的原理进行图像重建。从此以后，图像超分辨率重建技术进入一个崭新的研究发展阶段。Nguyen[16]等人首先采用小波方法进行图像超分辨率重建，取得了比较满意的成果。Irani 等[18]人提出的迭代反投影算法（IBP），通过计算不断的迭代模拟低分辨率图像和实际测试输入图像的误差来不断更新对重建图像的估计，但是解不唯一，没有引入先验知识约束。Stark 和 Oskoui[19]首次提出了凸集投影方法（POCS），此方法是基于理论集合的模型，取得了较满意的重建效果。目前图像超分辨率研究中贝叶斯法研究的最多，重建效果也较好。Schultz 和 Stevenson 等[20]人首次提出用最大后验概率估计（MAP）方法进行图像超分辨率重建，并用 Huber-Markov 随机场作为先验知识，取得较满意的效果。Elad[21]等人提出了混合 MAP 和 P OCS 的方法，基于统计理论和集合论，在图像重建的过程中不断地进行迭代优化，结合图像的先验知识和凸集约束特征。Babacan 提出用分层的贝叶斯架构模拟未知的高分辨率图像和运动参数，可以减

小估计误差，扩展性强。这类方法若有合理的先验知识，就可以获得非常理想的图像重建效果，但缺点在于计算量较大，不适合实时性要求高的场合。另外，输入的图像分辨率较低或者方法倍数较大，图像重建效果就不好。

后来，基于学习的图像超分辨率重建算法被提出，并迅速成为研究的热点，这类算法总的思想是，通过包含高分辨率图像和对应低分辨率图像的训练集建立学习模型，获取先验知识，指导重建高分辨率图像，重建图像的效果在很大程度上取决于训练集的规模和质量[22]。学习算法可以利用大量现存自然图像之间的统计关系来解决图像超分辨率的病态求逆问题，为图像超分辨率重建开辟了新的道路，还克服了基于重建的方法在分辨率提高倍数方面的局限性，近年来发展很快，成为研究的热点。A Hertzmann 等[23]人提出了基于多尺度（multiscale）自动回归的图像类比技术。Freeman WT[24]等人提出一种基于例子的学习算法，首先用很多自然图像构建训练图像集，然后通过标准的马尔科夫（markov）网络进行图像重建。Yang 等[25]通过构建过完备字典，提出基于稀疏表示的图像重建算法。但是主要适用于一般图像的超分辨率重建，不是专门针对人脸图像的算法。

针对人脸图像，最早是由 Baker 等 [26]等人提出了人脸图像超分辨率重建技术，即人脸超分辨率（Face Hallucination），也称为虚幻脸重建，使人脸图像超分辨率重建从图像超分辨率重建技术中单独分离出来，成为一个独立的研究领域。目前来说，人脸超分辨率重建主要是基于学习类的算法。他们利用高斯金字塔以及拉普拉斯金字塔，建立人脸图像的特征空间。通过学习得到映射关系，并以此作为先验知识，在贝叶斯框架下得到超分辨率重建人脸，缺点是人脸图像的一些部位会存在较大的错误或噪声。Liu[27]等人首次提出了基于两步框架的人脸超分辨率算法，他们给出了人脸超分辨率重建的三条约束准则，并结合全局参数模型与局部马尔科夫随机场的非参数模型来重建高分辨率人脸图像[12]。后来学者提出很多算法都是基于这种两步法框架的。Wang 等[28]人利用主成分分析（PCA）方法，认为输入的人脸图像可以看做是由一系列低分辨率图像线性组合得到的，利用 PCA 变换得到测试图像在训练人脸图像上的权重系数，结合权重系数和对应位置的高分辨率图像，重建出最后的高分辨率图像。Liu 等[29]提出一种基于多尺度和多方向特征的人脸超分辨率算法，利用金字塔学习人脸图像的低层次局部特征的空间分布，并结合塔状父结构和局部最优匹配算法来预测最佳先验模型 [12]。David Capel 等[30]将人脸划分成眼、鼻、嘴和面颊等不同部位，利用主成分分析 PCA 模型来描述不同部位的特征，由不同部位的权重系数加权重建得到最后的人脸图像[31]。Yang 等[32]提出一种利用相关过完备字典的稀疏表示的图像超分辨率算法，来增强人脸的局部细节信息。Zhuang 等[31]提出局部保持投影（LPP）和邻域残差补偿相结合的方法，先利用 LPP 提取图像特征，结合径向

基函数回归（RBF）进行重建。基于学习的方法属于新兴的算法，正在被广泛的研究，这种方法可以弥补基于重建的方法的很多不足，通过利用训练图像库的先验信息来指导图像重建，在放大倍数很大的情况下，仍然能保持较好的重建效果，是值得进一步研究和发展的方向[33]。

目前，图像超分辨率重建技术要想在现实应用中取得较好的效果，还面临许多的问题和挑战[34]。图像配准在序列图像超分辨率重建中占有很重要的地位，在很大程度上影响高分辨率图像重建的效果。以此同时，图像退化模型在实际应用中是很复杂的，会受到模糊、噪声、几何运动等很多因素的影响，目前现有的图像退化模型是不准确的，并且也难以对退化模型的参数进行精确的估计，这也会很直接地影响超分辨率重建图像的效果。还有，基于学习的重建算法，如何选取恰当的训练样本数量和质量是非常重要的。最后，衡量超分辨率重建图像质量的标准也是一项值得研究的课题。

1.2.2 低分辨率图像退化模型

在现实生活中受到成像硬件系统分辨率的限制和各种成像环境因素（如运动形变、光学模糊、噪声等）的干扰，获得的图像通常都不是严格意义上的高分辨率图像。低分辨率图像退化模型如图 1.3 所示。

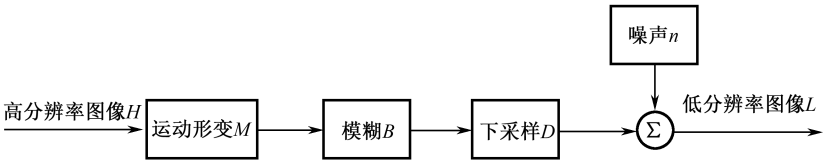


图 1.3 退化模型

图像退化模型包括四个部分，即运动形变、模糊、下采样和噪声，使得原始的高分辨率图像经过退化模型后，得到一幅或多幅低分辨率图像。下面对造成图像分辨率下降的几种因素进行简单的介绍。

（1）运动形变：运动形变的产生是由于真实景物与成像设备之间的相对运动，包括图像的全局运动和局部运动 [12]。全局运动是指整个图像空间的像素具有相同的运动特性，其运动参数可以用垂直和水平方向的位移来表示。局部运动是指图像空间中的多个物体具有各自的运动特性和参数。

（2）模糊：引起模糊的主要原因是光学模糊，用点扩散函数的卷积来描述，

在超分辨率重建的过程中，需要采用一定的手段对点扩散函数进行确认，然后通过对其函数进行反卷积计算实现重建。

（3）下采样：下采样会使图像丢失部分信息，降低图像的分辨率。

（4）噪声：成像过程中会加入很多种类型的噪声，通常为了降低问题求解的复杂性，假定噪声是独立同分布的高斯噪声。

按照图像退化模型，通常可以认为 n 幅低分辨率图像是由一幅高分辨率图像经过图像退化模型得到的，图像退化模型的数学公式表示为：

$$L_k = D \times B_k \times M_k \times H + n_k \qquad (k = 1, 2, \cdots, n) \tag{1.2.1}$$

式中，H 是高分辨率图像；L 是低分辨率图像；B 是模糊矩阵；M 是运动形变矩阵；D 是下采样矩阵；n 是噪声。

图像超分辨率重建的目的是在已知一幅或多幅低分辨率图像以及假定的图像退化模型的情况下，求高分辨率图像的过程。这个过程看似只是对图像退化模型进行求逆运算，但超分辨率重建是一个高度病态的问题，参数一般是很难估计的。这使得在实际应用中，误差是不可避免的，在恢复高分辨率图像过程中具有不稳定性，影响鲁棒性。在解的稳定性、收敛性和可靠性得不到保证的情况下，通常只能利用某些约束条件将问题进行转化，然后通过最优化的方法估计出最优解。

1.3　空间角度的人体行为识别介绍

计算机视觉技术属于计算机智能的一部分，它包含图像处理、模式识别等热点研究领域。与人类一样，来自视觉的信息占信息量的比重很大，因此计算机视觉研究的重要性不言而喻。对计算机视觉的研究并不是在于让计算机能和人眼一样看见外界环境，其最终目的是让计算机能对获取的外界信息进行智能分析与处理，模拟人脑功能，实现对现实世界的认知。计算机视觉是一门综合性很强的学科[35]，它涉及生物学、物理学、神经学、认知心理学、图像处理、模式识别、计算机科学等，这决定了计算机视觉技术拥有广泛的应用领域以及广阔的发展前景。在计算机视觉诸多应用领域中，应用于安防事业[36]的智能视频监控系统也占有重要的地位。

随着社会的发展，人们对于自己所生活的社会环境安全的需求日益增长，特别是公众对自身的人身安全以及财产安全得到保障的需求日益凸显，这使得世界

各国对社会的安防建设事业越来越重视。纵观近年来世界各国频繁发生的特大恐怖袭击事故，这些重大事件都突显出社会迫切需要提高预防公众安全事故发生的能力。

2014 年 3 月 1 日 21 时 20 分，在中国云南省昆明市火车站发生一起“昆明火车站暴恐案”，这是一起震惊中外的极其恶劣的暴力恐怖袭击案件。该案件是由新疆分裂势力组织策划的任意砍杀民众事件，一伙统一着装的歹徒手持凶器冲进昆明火车站广场、售票厅等，见人就砍，场景十分凶残。据统计，截至 3 月 2 日 18 时，已造成 29 死 143 伤。

2013 年 10 月 28 日中午时分，在北京天安门金水桥边发生一起“暴力恐怖袭击案”[37]。不法分子驾乘一辆吉普车闯入长安街，沿途快速行驶，故意冲撞游客群众，并最后撞向金水桥护栏，点燃车内汽油导致车辆起火燃烧，据统计该事件造成 5 人死亡，40 人受伤。

2012 年 12 月 14 日，在美国康涅狄格州的桑迪·胡克小学[38]发生“校园枪击惨案”。一名暴徒手持枪械闯入桑迪·胡克小学后，便直接向在校的教师及小孩肆意开枪，该事件共造成包括枪手在内的 28 人丧生，其中 20 名都是无辜的儿童，场面极其残忍。此事件也是美国历史上死伤最惨重的校园枪击案之一。

上述几个事件突出说明了我们社会迫切需求能有一套有效的措施来防止公众的人身安全及财产安全等遭受侵害。如今社会发展迅速，视频监控已经在社会的各个行业和领域随处可见，如图 1.4 所示。目前在经济社会生活中，对于安全事故的防范以及安全事故发生后的侦破，视频监控[39,40]有重要的应用价值，对维护国家及公共安全有重要的现实意义。

然而，当前视频监控系统只是停留在被动的数据采集阶段，系统功能单一，智能化程度低。大多时候还只是对某些特定环境下运动目标的检测以及跟踪，对目标做进一步的识别和理解分析工作还很欠缺。当安全事故已经发生后，通过视频监控，人工地进行侦破。而令人头疼的是，视频监控收集的数据庞大，因此需要耗费极大的人力，效率低下，检测效果也令人担忧。而另一方面，我们不仅需要通过视频监控帮助事后的侦破，更需要通过视频监控实时地进行安全事故的防范，这才是人们日常生活中所迫切需求的。

因此，针对以上背景提出了对视频中行人的动作理解与分析[41-43]以及异常行为实时检测的研究[44,45]。该技术将提高视频监控、安防系统等智能化程度，以便于能够充分发挥视频监控系统的主动监督作用、及时报警预防安全事故发生的作用。

图 1.4　经济社会生活中随处可见的视频监控

1.3.1　国内外研究现状

21 世纪以来，智能视频分析及其相关技术受到世界各国安防部门、国防建设、公司、研究所、高校等机构和学者们的极大关注。对视频中的信息进行分析与理解等研究在过去时间里迅猛发展，形成了比较全面的研究内容，相继取得了一系列阶段性的成果。许多国家有关部门以及领域内学者们对该项课题进行了大量的研究，下面将分别从国外与国内视角进行梳理。

1997 年，美国国防部高级研究项目署设立了由卡内基梅隆大学（CMU）领导的、麻省理工学院（MIT）、Sarnoff 公司等单位参与的智能视频监控系统重点项目（Visual Surveillance and Monitoring，VSAM）[46]。该系统不仅能用于民用监控，还可应用于军事领域，如军事基地、敌方战地、难民流动区域等的实时监控，有助于战场势态分析、为军事行动做出决策提供帮助。

1999 年，美国马里兰大学（Maryland）研制了一套实时视觉监控系统，命名为 W4 系统[47]，它不但可以跟踪行人并检测出行人的各个部位，而且通过建立各个部位的外观模型实现对多人的监控，在室内及室外等某些场景下实现了检测

人体的简单行为。

除此之外，还有麻省理工学院开发的人体运动跟踪系统 Pfinder，美国的 ObjectVideo 公司开发的视频分析系统，以色列的 NiceVision 公司的视频分析仪等。

该研究领域还有许多国际上权威的学术期刊，如 PAMI（IEEE Transactions on Pattern Analysis and Machine Intelligence）、IJCV（International Journal of Computer Vision）、IVC（Image and Vision Computing）等，以及重要的学术会议，如 CVPR（IEEE Conference on Computer Vision and Pattern Recognition）、ICCV（IEEE International Conference on Computer Vision）、ICPR（International Conference on Pattern Recognition）等。

在国内，该课题的研究也得到了国家、科研院所、高校以及专家学者的高度重视。中科院自动化研究所的模式识别国家重点实验室[48]（Center for Biometrics and Security Research, CBSR）开发了一套针对室内、室外环境监控的智能视频检测系统，该系统能实时记录和分析场景中目标的姿态、行为等，并且支持对运动目标执行多生物特性的识别。此外，他们还建立了数个行为识别的数据库，为国内研究提供了很好的基础和平台。相继投入研究的还有清华大学、北京航空航天大学、浙江大学、西北工业大学、微软亚洲研究院等。在 2008 年北京奥运会期间，中国在安防系统中也首次采用了该领域最新的技术，提供人脸识别和视频报警等重要功能，预防各种安全突发事件，为北京奥运会的顺利召开提供了坚实的安全保障。

1.3.2 人体行为视频数据库

对于人体行为识别的研究，比较繁重的一项工作是需要采集人体行为的视频数据库。随着人体行为识别研究的不断推进，领域里已经出现了一些较为权威的、完整的、被多数研究所采用的行为视频数据库。在一定程度上可以说，行为数据库为该课题的研究提供了必要的数据依据。下面将介绍几个公开的、使用较为广泛的行为数据库。

1．Weizmann 人体行为数据库

Weizmann 人体行为数据库是以色列 Weizmann 科学院采集的，其主要特点是采集视频数据的背景、拍摄视角都是静止不变的。该视频库总共有 90 个视频片段，由 9 个不同性别、不同体型的人分别进行 10 种不同的动作，包括行走（Walk）、跑步（Run）、跳跃（Jump）、侧身行走（Gallop sideways）、弯腰（bend）、

挥单手（One-hand wave）、挥双手（Two-hands wave）、原地跳（Jump in place）、挥臂原地跳跃（Jump jack）、单腿跳（Skip），每个视频段只存在一个人和一种行为。图 1.5 所示是该视频数据库的部分示例图。

（a）行走　（b）跑步　（c）跳跃

（d）侧身行走　（e）弯腰　（f）挥单手

（g）挥双手　（h）原地跳　（i）单腿跳

图 1.5　Weizmann 人体行为数据库部分示例图

2．KTH 人体行为数据库

KTH 人体行为数据库是皇家理工学院采集的，其特点是在多个场景下拍摄，背景相对静止，某些场景拍摄的镜头有拉近拉远操作。该数据库由 25 个不同性别、不同体型的人，分别执行 6 种行为，包括行走（Walking）、拳击（Boxing）、挥手（Hand Waving）、拍手（Hand Clapping）、慢跑（Jogging）、跑步（Running），并且每个人每种行为分别在四种场景（户外、户外镜头变焦、户外不同着装、室

内）下依次采集，每个视频只存在一个人和一种行为。图 1.6 所示是该视频部分示例图。

（a）行走

（b）挥手

（c）慢跑

（d）拍手

（e）跑步

（f）拳击

图 1.6　KTH 人体行为数据库部分示例图

3．中科院人体行为数据库[48]

中科院自动化研究所也建立了一个人体的步态数据库，其特点是拍摄视角有斜视、俯视、平视三种，而且包含单人行为以及多个人交互行为。该视频库总共1446 个视频片段，拍摄场景都在室外。单人行为有行走、跑、跳跃、弯腰等 8 种，多人交互行为有抢劫、打斗、尾随等异常行为。该行为数据库不是公开的数据库。

本章参考文献

[1] 山世光. 人脸识别中若干关键问题的研究[D]. 北京：中国科学院研究生院，2004.

[2] 张晓华, 山世光, 曹波, 等. CAS-PEAL 大规模中国人脸图像数据库及其基本评测介绍[J]. 计算机辅助设计与图形学学报, 2005, 17(1): 9-17.

[3] Yin Baocai. BJUT-3D Face Database [OL]. [2008-04-22] http://www.bjut.edu.cn/sci/multimedia/mul-lab/3dface/face_database.htm.

[4] The CUHK Face Sketch Database is available for download at: http://mmlab.ie.cuhk.edu.hk/facesketch.html.

[5] Belhumeur, P.N., Kriegman, D. What is the set of images of an object under all possible lighting conditions? IEEE Computer Society Conference on Computer Vision and Pattern Recognition[C]. San Francisco: IEEE, 1996, 270-277.

[6] Belhumeur P.N.，Hespanha J. P.，Kriegman D. J. Eigenfaces vs. Fisherface: Recognition Using Class Specific Linear Projection[J]. IEEE Trans. PAMY, 1997, 19(7), 711-720, July.

[7] Jones, M.J., Viola, P. A cluster-based statistical model for object detection. IEEE International Conference on Computer Vision[C].Kerkyra: IEEE, 1999, 2:1046-1053.

[8] V.Blanz, T.Vetter. A morphable model for the synthesis of 3D faces. Proc. of SIGGRAPH'99[C]. Los Angeles, 1999, 187–194.

[9] T.Vetter, V.Blanz. Estimating coloured 3d face models from single images: An example based approach. European Conference on Computer Vision(ECCV) [C].Freiburg, 1998, 499-513.

[10] Cortes C, Vapnik V. Support-vector networks[J]. Machine Learning, 1995, 20(3):273-297.

[11] 陈文. 小训练样本集下人脸图像超分辨率重构算法研究[D]. 湖南：中南大学,2010.

[12] 沈华. 基于插值和主元素分析的人脸超分辨率算法研究[D]. 湖南：湖南大学,2010.

[13] 杨宇翔.图像超分辨率重建算法研究[D]. 安徽省：中国科学技术大学, 2013.

[14] J Allebach, P W Wong. Edge-directed interpolation[J]. IEEE International Conference on Image Processing, 1996, 3: 707-710.

[15] Li X, Orchard M T. New edge-directed interpolation[J]. IEEE Trans. on Image Processing, 2001, 10(10): 1521-1527.

[16] N. Nguyen, P Milanfar. An efficient wavelet-based algorithm for image super-resolution[C]. IEEE International Conference on Image Processing, 2002, 2: 351-354.

[17] Huang TS, Tsai R. Multi-Frame Image Restoration and Registration[J]. Advances in Computer Vision and Image Processing,1984, 1(2): 317-339.

[18] Irani M, Civanlar M R. The Feasible Solution in Signal Restoration[J]. IEEE Transactions on Acoustics, Speech and Signal Processing. 1984, 3(2): 201-212.

[19] Stark H,oskoui P. High-resolution image recovery from image-plane arrays, using convex projection[J]. Journal of the Optical Society of America A,1989, 6(11): 1715-1726.

[20] R. R. Schulz, R.L. Stevenson. Extraction of high-resolution frames from video sequences[J]. IEEE Trans. on Image Processing,1996, 5: 996-1011.

[21] Elad M, Feuer A. Restoration of a single super-resolution image from several blurred, noisy and understand measured images[J]. IEEE Trans on Image Processing,1997, 6(12): 1646-1658.

[22] 朱婷婷. 图像插值超分辨率重建算法研究[D]. 四川. 西南交通大学, 2010.

[23] A Hertzmann, C E Jacobs, N Oliver. Image analogies[C]. The 28th Annual Conference on Computer Graphics and Interactive Techniques, 2001, 327-340.

[24] W T Freeman, T R Jones, E C Pasztor. Example-based super-resolution[J]. IEEE Computer Graphics and Applications,2002,22(2):56-65.

[25] J C Yang, J Wright, T Huang. Image super-resolution as sparse representation of raw image patches[J]. IEEE Conference on Computer Vision and Pattern Recognition, 2008: 1-8.

[26] Baker S, Kanade T. Limits on super-resolution and how to break them[J]. IEEE Trans. on Pattern Analysis and Machine Intelligence,2002, 24(9): 1167-1183.

[27] C Liu, H Y Shun, C H Zhang. A two-step approach to hallucinating faces: Global parametric model and local nonparametric model[J]. IEEE Computer Society Conference on Computer Vision and Pattern Recognition,2001, 1: 192-198.

[28] X G Wang, X O Tang. Hallucinating face by eigentransformation[J]. IEEE Trans. on Systems, Man and Cybernetics, Part C, 2005(3): 425-434.

[29] W Liu, D H Lin, X O Tang. Neighbor combination and transformation for hallucinating faces[J]. IEEE Conference on Multimedia and Expo,Amsterdam, 2005, 145-148.

[30] David C, Andrew Z. Super-resolution from multiple views using learnt image models[C]. In Proc,CCVPR,2001, 2(6): 27-34.

[31] Y Zhuang, J Zhang, F Wu. Hallucinating face: LPH super-resolution and neighbor reconstruction for residue compensation[J]. Pattern Recognition,2007, 40(11): 3178-3194.

[32] J C Yang, H Tang, Y Ma. Face hallucination via sparse coding[J]. IEEE Conference on Image Processing, 2008: 1264-1267.

[33] 乔建苹. 超分辨率重建与图像增强技术研究[D]. 山东：山东大学,2008.

[34] S. Derin Babacan. Variational Beyesian Super Resolution[J]. IEEE Transaction on image processing, 2011, 20(4): 984-999.

[35] Jungong Han, Ling Shao, Dong Xu, et al.. Enhanced Computer Vision With Microsoft Kinect Sensor: A Review[J]. IEEE Transactions on Cybernetics, 2013, 43(5): 1318-1334.

[36] Weilun Lao, Han J. Automatic Video-based Human Motion Analyzer for Consumer Surveillance System[J]. IEEE Transactions on Consumer Electronics, 2009, 55(2): 591-598.

[37] 高学敏. 中国反恐立法的形式、问题与对策[J]. 行政与法, 2014, 7: 126-129.

[38] 胥长寿. 校园暴力的现状、成因及预防[J]. 科学咨询（科技.管理）, 2014, 22(6): 18-20.

[39] Meghdadi A H, Irani P. Interactive Exploration of Surveillance Video through Action Shot Summarization and Trajectory Visualization[J]. IEEE Transactions on Visualization and Computer Graphics, 2013, 19(12): 2119-2128.

[40] Sayed M S, Delva J G R. An Efficient Intensity Correction Algorithm for High Definition Video Surveillance Applications[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2011, 21(11): 1622-1630.

[41] 徐光祐, 曹媛媛. 动作识别与行为理解综述[J]. 中国图象图形学报, 2009, 14(2): 189-195.

[42] Zhuolin Jiang, Zhe Lin, Larry S D. Recognizing Human Actions by Learning and Matching Shape-Motion Prototype Trees[J]. IEEE Transactions on PAMI, 2012, 34(3): 533-547.

[43] Panahandeh G, Mohammadiha N, Leijon A, er al.. Continuous Hidden Markov Model for Pedestrian Activity Classification and Gait Analysis[J]. IEEE Transactions on instrumentation and measurement, 2013, 62(5): 1073-1083.

[44] 朱旭东, 刘志镜. 基于主题隐马尔科夫模型的人体异常行为识别[J]. 计算机科学, 2012, 39(3): 251-275.

[45] Xiaogang Wang, Meng Wang, Wei Li. Scene-Specific Pedestrian Detection for Static Video Surveillance[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 36(2): 361-374.

[46] Collins R, Lipton A, Kanade T, et al.. A system for video surveillance and monitoring: VSAM final report[S]. Carnegie Mellon University, 2000.

[47] Haritaoglu I, Harwood D, Davis L S. W4 real-time surveillance of people and their activities[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2000.08, 22(8): 809-830.

[48] Cbsr, Center for Biometrics and Security Research [EB/OL]. http://www.cbsr.ia.ac.cn/china/Action%20Databases%20CH.asp, 2008.

第 2 章

人脸图像配准和人脸检测跟踪

本章根据人脸模型的 2D/3D 特性，提出了基于图像纹理特征点的 3D 人脸配准算法。根据 3D 人脸模型的特性，将 3D 人脸投影到 XOY 平面上，获得 3D 人脸模型对应的正面 2D 人脸纹理特征，之后利用 VOSM 算法检测 2D 人脸对应的特征点，再反投影到 3D 空间获得 3D 人脸特征点的位置，之后重建坐标系，建立新的三角网格，将 3D 人脸重建并实现配准。

接着，本章还提出结合均值粒子滤波器的人脸检测和跟踪算法。算法分为人脸检测部分和跟踪部分。人脸检测部分使用 Gentle AdaBoost 算法实现，并提出递进复杂度的级联分类器，增加人脸检测的效率。人脸跟踪部分，使用粒子滤波器算法实现，提出了均值权重粒子滤波器算法，解决了传统粒子滤波器高权重粒子产生的误差问题。最后将两部分结合，提出人脸检测校正策略，提升了人脸跟踪准确性，并解决了粒子滤波器样本匮乏的问题。实验表明新算法性能更好，保存的人脸截图冗余性低，出错率小。

2.1 人脸配准简介

人脸配准是指将人脸图像分辨率变为一致，并将各个关键特征部位的坐标位置尽量接近，如眼睛、鼻子、嘴巴等。人脸配准是人脸识别相关技术实现的基础，没有进行配准过的人脸库是无法直接用于人脸识别相关技术的。因此若数据库采

集后没有配准，则必须先进行配准。传统 2D 人脸配准的方法一般分为两种，一种是直接对特征点进行手工标定，得到各个特征点坐标，之后使用线性变换，得到配准图像，属于手动配准。另一种则是利用人眼检测等手段，寻找到图像特征点，再进行配准。这类方法一般情况下，需要少量的对人眼检测等进行一些手工校正，防止误检测，因此属于半自动配准的方法。2D 人脸一般信息量较少，使用特征点也不多，配准相对容易，配准主要难点在于，无法正确检测到特征点，配准算法本身已经比较成熟。因此本章重点研究了 3D 人脸数据库的配准实现，下面将详细地介绍 3D 人脸数据的配准方法。

2.1.1 3D 人脸配准简介

3D 人脸的相关技术在人工智能、虚拟现实等领域是一个十分热门的技术，相比 2D 人脸的相关技术发展已久，并且日趋成熟，3D 人脸的相关技术还处于起步阶段。对比 2D 人脸，3D 人脸的信息量更丰富，除了颜色信息、位置信息，3D 人脸多了深度信息，并且，3D 模型不同于 2D 图像，是由三角网格构成的，因此 3D 模型可以很轻易地实现模型形变，这样 3D 人脸模型就可以实现表情变化、姿态变化等，因此 3D 人脸识别对多表情、多姿态的识别[1-4]效果有很大提升。但是由于 3D 人脸是由三角网格构成的，也导致 3D 人脸的配准十分困难。对于 2D 人脸，只需要准确找到特征点位置，即可以进行有效配准。但是对于 3D 人脸，仅仅有特征点位置，还并不够。3D 人脸的配准需要让三角网格信息一致，并将特征点配准之后，才能用于下一步研究。现阶段，可以直接使用的 3D 人脸数据库不多，国内最著名的有北京工业大学的 BJUT-3D 数据库，国外则有 3D_RMA、GavabDB 等。但是这些数据库在公开的同时，并没有将 3D 人脸模型进行配准，无法直接使用。由于上述原因，本章对 BJUT-3D 人脸数据库进行了研究，对数据库进行了重建并配准。

2.1.2 数据库简介

BJUT-3D 是北京工业大学建立的中国第一个大型 3D 人脸数据库[5]，3D 模型使用 Cyber Ware 3030RGB/PS 激光扫描仪获取。3D 人脸模型获取时，要求目标正坐在椅子上，面朝前方，保证扫描出来的 3D 人脸为面朝正面人脸头像。为了保证获取过程中，不受光照等其他条件的影响，扫描仪所在房间是一个封闭的、

没有其他光源的房间。图 2.1 为模型采集示意图。

图 2.1　模型采集示意图

数据库共包括 500 个 3D 人脸模型，其中男、女各 250 名，年纪从 16～49 岁不等，所有人物都没有佩戴眼镜或者其他饰物。图 2.2 为采集到的 3D 人脸模型示意图。

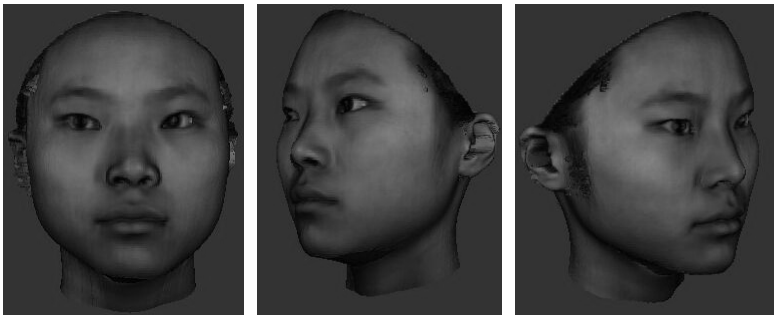


图 2.2　获取到的 3D 人脸示意图

2.2　3D 人脸配准

本节中，提出了基于 3D 人脸纹理图像的 3D 人脸配准算法。和 2D 人脸配准一样，3D 人脸配准也需要寻找特征点。但是，3D 人脸中，手动寻找特征点的

难度非常大，工作量太大，无法实现。3D 人脸特征检测的方法，也并不成熟，没有办法直接检测到 3D 人脸中的眼睛、鼻子等特征位置，因此，本章选择找到 3D 人脸模型对应的纹理图像，并对纹理图像进行特征检测，之后再找到 3D 人脸模型对应的特征位置，再对 3D 人脸进行重建，并配准。

2.2.1 获取纹理图像

纹理图像是指 3D 模型的对应平面图像，由于所获得的人脸模型都是默认的正面人脸，因此我们只需要将 3D 人脸投射到 XOY 平面，忽略 Z 坐标的深度信息，即可以获得对应 3D 人脸的正面人脸纹理图像[5,6]。为了保证所有的纹理图像大小一致，在进行纹理投射时，所得到的图像进行了归一化，均为 1101×901 的 2D 图像。在进行纹理投射过程中，会碰到许多（x,y）坐标信息一致，但是 z 坐标信息不同的点。实际使用中，我们只需要人脸表面的点作为纹理图像，因此需要设定深度信息 z 的阈值，过滤掉多余的点。对于 3D 人脸图像，越是靠近面部表面的点，深度值越小，因此设定阈值 z_θ，小于 z_θ 的点保留，大于 z_θ 的点则舍去。之后将所有留下的点投射到 XOY 平面，保留颜色信息，点与点之间的颜色使用线性插值的方法进行填充，即可得到的正面人脸纹理图像。图 2.3 为得到的正面人脸纹理图像示意图。

图 2.3　正面人脸纹理示意图

2.2.2　检测特征点

得到 2D 人脸纹理图像后，我们就可以使用 2D 人脸图像特征点检测方法，对图像进行检测，获得特征点。由于人脸纹理图像的图像内容本身比较简单，因此我们可以采用比较方便的手段获得人眼位置以及人脸位置，例如，本章中，使用 Matlab 工具中的人眼滤波器，得到了人眼的位置，然后使用了边缘检测，得到了人脸的大概位置。经过测试，500 幅图像进行检测，只需要 5 分钟不到的时间就可以完成，并且准确度超过 90%，只有少部分的图像需要手动微调，效率较高，可以满足需求。图 2.4 为检测结果示意图。

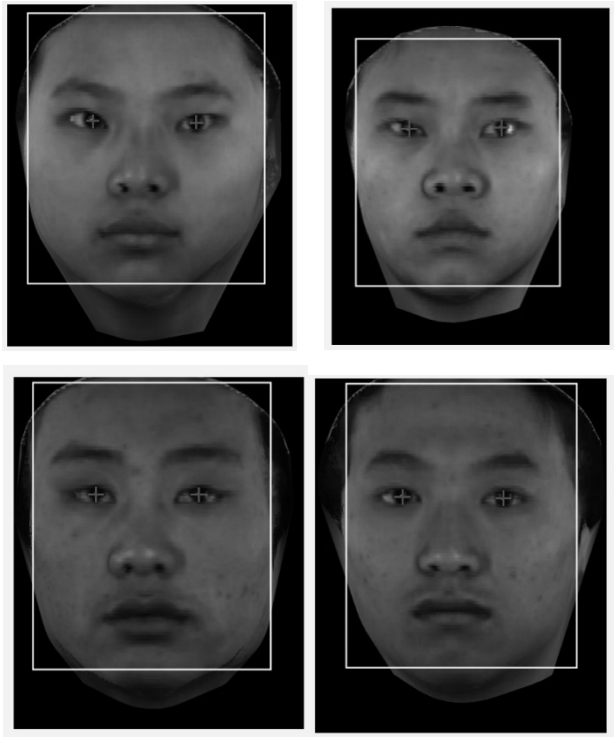


图 2.4　人眼定位及人脸定位结果示意图

2.2.3　细化特征点位置

在 2.2.2 节中，对人眼以及人脸位置进行了定位，这个步骤可以认为是特征

点的粗定位，仅仅使用这些特征位置，还无法完成配准，需要对人脸其他进行细节定位，才能实现下一步。因此特征点粗定位后，需要再进行特征点的精确定位。对人脸特征精确定位的方法一般采用 ASM 算法，ASM 算法原理如下。

1．形状模型的建立

对于 N 个训练样本，获取训练数据

$$X_i = \{(x_{i1}, y_{i1}, x_{i2}, y_{i2}, \cdots, x_{in}, y_{in})^T\}_{i=1}^{N} \tag{2.2.1}$$

其中，n 为特征点数量，(x_{ij}, y_{ij}) 为第 i 幅图像的第 j 个特征点的坐标，一般使用 98 个特征点进行面部描述，即 n=98。

2．局部灰度模型的建立

对于第 j 个特征点, $j = 1,2,3,\cdots,n$，在第 i 幅图像上，$i = 1,2,3,\cdots,N$，以该点和最相近的其他两点连线，并在连线的垂直方向上，取特征点两边的 k 个像素灰度值,之后可以得到灰度值向量，向量维度为 $2k+1$，如式（2.2.2）所示：

$$g_{ij} = (g_{ij1}, g_{ij2}, \cdots, g_{ij(2k)}, g_{ij(2k+1)})^T \tag{2.2.2}$$

对其求差分后，有式（2.2.3）：

$$\mathrm{d}g_{ij} = (g_{ij2} - g_{ij1}, g_{ij3} - g_{ij2}, \cdots, g_{ij(2k+1)} - g_{ij(2k)})^T \tag{2.2.3}$$

标准化后，得到式（2.2.4）：

$$y_{ij} = \mathrm{d}g_{ij} \Big/ \left(\sum_{l=1}^{2k} |\mathrm{d}g_{ijl}| \right) \tag{2.2.4}$$

3．获得局部灰度特征模板

最后通过式（2.2.4），可以获得每个特征点 j 对应的灰度特征模板，即可以对图像进行形状匹配，式（2.2.5）为最后的灰度特征模板的表达式。

$$\overline{y_j} = \frac{1}{N}\sum_{i=1}^{N} y_{ij}, S_j = \frac{1}{N}\sum_{i=1}^{N} (y_{ij} - \overline{y_j})(y_{ij} - \overline{y_j})^T \tag{2.2.5}$$

本章中借鉴了 VOSM 算法[7]对人脸特征精确定位，VOSM 算法为 ASM 算法的改进算法。图 2.5 为 VOSM 算法对人脸特征细定位的结果示意图。

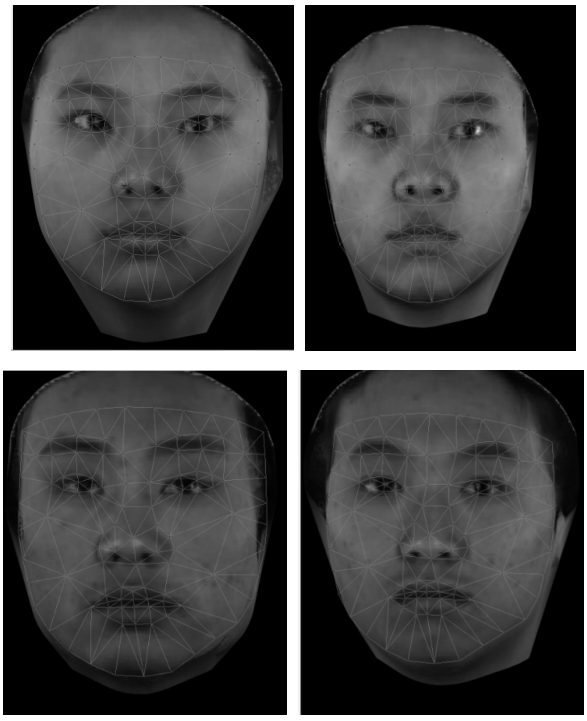


图 2.5　VOSM 特征点检测示意图

2.2.4　特征点模型标准化

为了让重建后的3D人脸模型和一般标准3D人脸模型有同样的结构，本章使用了 3D 人脸模型中最为常用的 Candide 3 模型[8]。Candide 3 模型是一个国际常用的 3D 标准人脸模型，模型对应的形变内容十分齐全，包括各类表情变化的动画等，泛用性极高。Candide 3 模型由 113 特征点组成，本章中选取了其中 71 个特征点来描述人脸图像。在获取了 VOSM 特征点后，建立 VOSM 模型和 Candide 3 模型之间点的映射关系，得到 Candide 3 模型对应的特征点位置，图 2.6 即为 Candide 3 模型示意图。

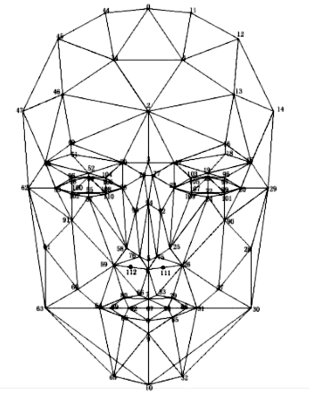


图 2.6　Candide 3 模型示意图

2.2.5 3D 人脸模型配准

在得到所有特征点后，就可以对 3D 人脸模型进行配准。和 2D 图像不同的是，2D 人脸的配准可以通过对图像的形变直接得到，但是 3D 人脸为了保证三角网格相同，需要以特征点为基准，重新建立坐标系，并对三角网格进行重建，这样才能保证所有的 3D 人脸特征点坐标位置，三角网格结构和顶点位置一致。为了实现这个目标，需要 2D 特征点对应在 3D 人脸模型的深度值，找到在 3D 人脸模型对应的位置。对于所得到特征点，已经得到其在纹理图像中的（*x,y*）的坐标信息，因为纹理图像和 3D 图像的映射关系是已知的，所以可以直接得到对应特征点的（*x,y,z*）的信息。但是虽然 VOSM 算法对人脸特征点的定位已经较为准确，但是再投射到 Candide 3 模型和 3D 人脸的过程中，依旧会产生误差，虽然可能（*x,y*）的差距不大，但是反应到对应的 3D 模型中，*z* 的偏差会比较明显，特征点可能不在人脸所在的平面上，因此这部分需要进行一些手工微调，让 3D 人脸特征点位置更加准确。图 2.7 为 3D 人脸模型中特征点定位完成后的示意图。

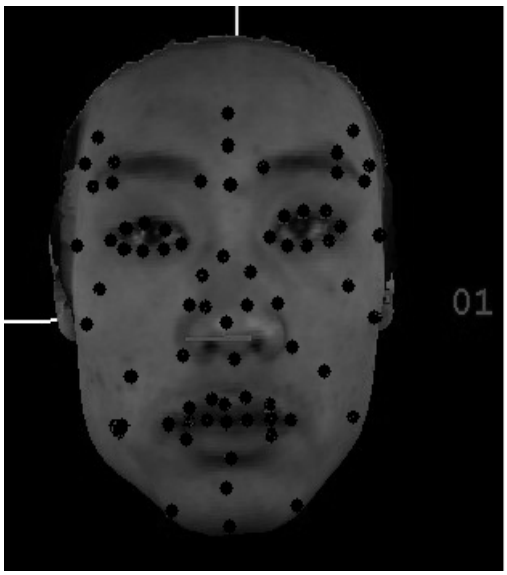


图 2.7　3D 人脸特征点定位

得到所有 3D 人脸的特征点的坐标后，就可以对模型进行重建。首先进行坐标系重建，本章中新的坐标系以两眉间的中点作为坐标系原点，建立坐标系，其他点的坐标可以由原坐标系和新坐标系的对应关系计算得到。之后进行三角网格的重建，对于 3D 模型，都是使用三角网格进行构建，因此可以将所有特征点相

连，形成粗略的三角网格，之后对网格进行细化获得需要的模型。细化方式具体如下：

（1）取三角网格对应的三角形各边的中点，并计算中点的空间坐标 (x_m, y_m, z_m)。

（2）寻找和这些中点具有相同的（x_m, y_m）信息的面部点，若有则进入（3），若无进入（4）。

（3）设对应面部点的空间坐标为（x_m, y_m, z_f），其中 z_f 为对应面部点的深度值，并获得其对应的颜色信息。，

（4）若（x_m, y_m）处没有对应面部点，则寻找和（x_m, y_m）在 XOY 平面投影中距离最小的面部点，设其坐标为（x_f, y_f, z_f）。其中（x_f, y_f）为面部点对应的平面坐标值，并满足式（2.6）。

$$(x_f, y_f) = \arg\min \sqrt{(x_m - x_f)^2 - (y_m - y_f)^2} \tag{2.2.6}$$

（5）将新找到的这些面部点和原本的特征点再相连，组成新的三角网格。

细化步骤流程图如图 2.8 所示。

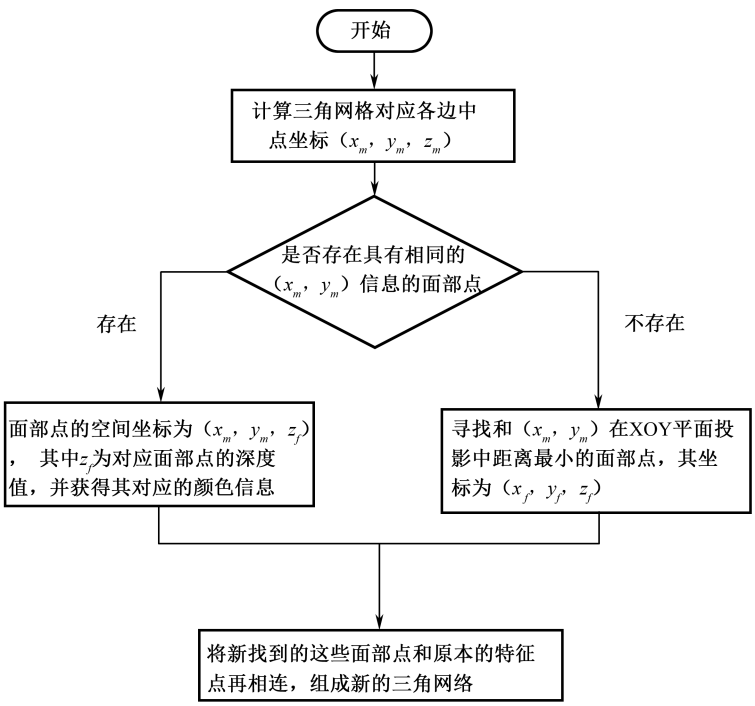


图 2.8　三角网格细化步骤流程图

重复以上步骤，就可以对整个面部细化，得到重建的人脸。细化 3D 网格，本身十分耗费时间，同时 3D 网格过于密集，顶点过多，也会导致模型的使用性下降。基于这个原因，采用的分块细化方法，即将人脸划分为左面颊、右面颊、左眉、右眉、左眼、右眼、前额、鼻子、嘴、下巴共 10 个部分，每部分的细化次数从 3 次到 5 次不等，如眼睛、鼻子等复杂部位细化次数多，而面颊这类简单的部位细化次数则较少。

这样一来，重建的 3D 人脸就保证了特征点一致，网格信息一致，并且保留了面部需要的有效信息。图 2.9 为图 2.7 对应的重建的 3D 人脸模型。

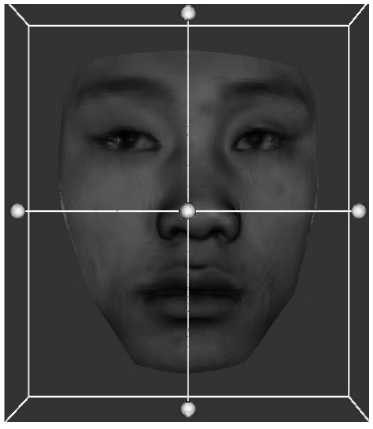


图 2.9　重建的 3D 人脸模型示意图

2.3　人脸检测简介与常用算法介绍

人脸检测是指对输入的图像或者视频帧中，确认是否有人脸的存在，若有则将人脸位置框出的过程。人脸检测是人脸识别技术的基础，也是人脸识别系统中的第一步，因此其重要程度也是最高的。一般来说，人脸检测要求尽可能地找出图像或者视频中的各类人脸，包括各类姿态、表情、饰物，以及光照条件的人脸。能检测的各个类型的人脸种类越多，人脸检测算法的鲁棒性则越好。

人脸检测在模式识别[9]中，是一个二分类问题，即这个问题的判断结果只有是或者不是两种可能。一般情况下，人脸检测可分为两大类：非学习类检测方法和学习类检测方法。

非学习类的检测方法主要是对人脸进行几何分析、颜色分析、运动分析等，针对人脸的形状、肤色以及运动轨迹等进行检测，常用的方法包括 Gobor 图形检测，肤色检测等，这类算法的优点在于得到分类器的速度很快，方法简单、易用。但是缺点也是明显的，这类算法的检测效果非常粗糙，检测率很不理想，针对类人的物体无法分辨，因此并不是人脸检测的主流算法。

学习类的算法即机器学习的方法，机器学习是研究如何使用机器来模拟人类学习活动的一门学科。机器学习是一门研究机器获取新知识和新技能，并识别现有知识的学问。这里所说的“机器”，指的就是计算机、电子计算机、中子计算机、光子计算机或神经计算机等。其过程主要包括以下几步：

（1）将样本分为两类，一类正样本，一类负样本。人脸检测中，即人脸和非人脸。

（2）寻找样本的特征，以这个特征来对样本进行分类，得到分类器，之后用分类器对样本进行测试，得到检测率。

（3）对各类特征进行循环处理，对特征进行组合，直到得到最好的检测率，并得到最终的分类器。

可以看出，机器学习的过程中，会不断告诉计算机，如何对样本进行分类是正确的，之后不断修正分类器，这个过程称为“学习”。学习类算法可以再细分为两类，一类是非监督训练的学习算法，一类是有监督训练的学习算法。非监督训练的学习算法是指学习过程中，每一轮学习中错误分类的样本和正确分类的样本同等看待，并不区分。监督训练的学习算法则不同，在每一轮学习过程中，会更多地注意上一轮错分的样本。

人脸检测发展到现在，基本上有效的算法都是有监督训练的学习算法，这也是因为这类算法更符合人类的学习过程。人脸检测算法非常多，不能一一列举，因此下面将简单介绍目前最常用的人脸检测重要算法。

2.3.1　神经网络

人工神经网络就是模拟人的思维。这是一个非线性动力学系统，其特色在于信息的分布式存储和并行协同处理。虽然单个神经元的结构极其简单，功能有限，但大量神经元构成的网络系统所能实现的行为却是极其丰富多彩的。

神经网络利用众多的节点（也称为神经元）形成一个网络，将众多的图像样本及鉴定结果输入，使网络渐渐获得可以鉴定类似样本的能力。其优点比较明显：

（1）在非线性分类中可以充分逼近理想结果。

（2）所有定量或者定性的信息都保存在了各个网络的神经元节点上，因此算法具有较好的鲁莽性。

（3）拥有较好的自适应性，可以适用于不同的环境。

神经网络是检测效果十分好的检测算法，但是其缺点也是明显的，神经网络的分类器获取复杂，对多姿态人脸容忍度非常低，同时神经网络的检测效率是所有人脸检测算法中偏低的，因此不能用于视频中的实时检测，无法作为实时人脸检测系统中的算法。

2.3.2 支持向量机（SVM）

支持向量机于 1995 年由 Vapnik 和 Corinna Cortes 等人所提出，建立在统计学习理论的 VC 维理论和结构风险最小原理基础上的，根据有限的样本信息在模型的复杂性（即对特定训练样本的学习精度）和学习能力（即无错误地识别任意样本的能力）之间寻求最佳折中，以求获得最好的推广能力。

SVM 算法提出时是用于线性可分的模型进行分类的算法，经过发展后，后提出对于线性不可分的模型，可以采用将低维向量投射到高维，建立超平面，将问题转化为线性可分的问题，从而使得高维特征空间采用线性算法对样本的非线性特征进行线性分析成为可能；之后 Osuan 等人首次使用 SVM 算法在人脸检测中使用，对 19×19 大小的人脸进行检测，取得良好的效果。但是 SVM 和神经网络有着类似的问题，获取的分类器由于是在高维空间获取的，因此分类器维度很高，训练过程也相对复杂，无法实现实时检测的要求。

2.3.3 AdaBoost 算法

AdaBoost 算法是典型的机器学习类算法，其全称为 Adaptive Boosting，在 1995 年由 Freund 和 Schapire 在首次提出，并在 1997 年 Viola 等第一次将 AdaBoost 算法首次应用在人脸检测中，取得非常好的效果。AdaBoost 算法使用 haar 特征区分人脸与非人脸的特征，并提出利用积分图快速计算 haar 特征，作为弱分类器使用，保证了算法的效率。同时该算法是有监督的机器学习算法，每次迭代后，弱分类器都会对训练反馈，让下一轮迭代过程更加注重前轮无法分类的样本。这样，随着迭代次数的增加，理论上可以达到很高的检测率。为了进一

步提高算法的检测效率，还提出了级联式分类器的模型，达到了实时检测的需求。

AdaBoost 算法的缺点是训练耗时，且需要大量训练样本的收集，特别是负样本。同时分类器依赖训练样本的类型，若训练时没有加入正面人脸以外的其他类型的样本，如多姿态人脸，分类器在检测时对此类样本的鲁棒性就会比较低，但是在训练时添加对应样本，则可以增强对此类样本检测的鲁棒性。之后，AdaBoost 算法也出现许多改进方法，本章中所使用的 Gentle AdaBoost 人脸检测算法就是其中之一，后文中会详细介绍该算法。

2.4　Gentle AdaBoost 人脸检测算法

基于上文的研究基础，本节实现了 Gentle AdaBoost 算法，并提出了递进复杂度的级联分类器，AdaBoost 算法的核心思想是寻找弱分类器，然后利用训练过程构造强分类器进行检测。Gentle AdaBoost 的核心思想和传统的 AdaBoost 类似，只是在构造弱分类的方法上进行了优化。

2.4.1　图像训练预处理

首先需要收集正样本的图像和负样本的图像，本章中正样本的图像使用 CAS-PEAL-R1 的人脸数据库来获取，以下简称 CAS 数据库。由于 CAS 数据库的图像照到了脖子以上，不能直接用于人脸检测的训练，因此将图像截取到标准人脸，并将图像配准，变为 20×20 大小，如图 2.10 所示。

（a）CAS 数据库原图像

（b）处理后的图像

图 2.10　CAS 数据库预处理

非人脸则直接使用没有人脸的图像进行截取，同样要保证大小为 20×20，但是由于需要训练级联分类器，因此需要的非人脸数量远远大于人脸数量。因此，笔者训练使用了超过 80000000 张非人脸。在训练中，使用的图像都为灰度图像，因此图像收集后都要进行灰度化。

2.4.2 haar 特征选择和积分图的计算

haar 特征[10]是图像处理中常用的特征提取方式，在 Gentle AdaBoost 中作为弱分类使用。图 2.11 展示了 5 种基本 haar 特征，其边长单位即为像素。

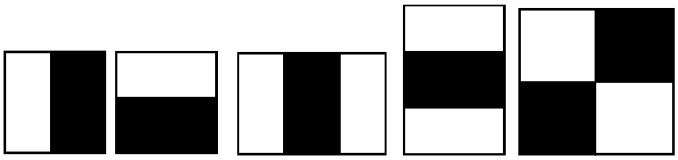


图 2.11　基本 haar 特征

其中，图 2.11（a）则表示这个基本 haar 特征的基础模型为 1 行×2 列的像素块，之后可以将这个基础模型放在 20×20 窗口的任意位置，以任意比例放大（不能超出窗口本身）进行放置，对应的所有这样的特征，均为（a）类 haar 特征，其余类的 haar 特征以此类推可以得到，以这个方式在 20×20 窗口中共可以得到 78460 个 haar 特征。对于每一个 haar 特征，其特征值是由一个特征白块的像素和值减去黑块的像素和值，为了快速计算 haar 特征值，我们引入积分图这一概念。

积分图概念源自积分运算，其表达式如式（2.4.1）所示：

$$f(x,y)=\int_0^y\int_0^x g(x',y')\mathrm{d}x\mathrm{d}y \tag{2.4.1}$$

其中 $f(x,y)$ 表示积分图，$g(x',y')$ 表示原图像，其计算方式就是将位于 (x,y) 的左上角所有像素值的和作为积分图对应 (x,y) 处的积分图值。因此我们可以将式（2.4.1）化简为式（2.4.2）：

$$f(x,y)=\sum_{x'\leqslant x,y'\leqslant y} g(x',y') \tag{2.4.2}$$

其中 $f(x,y)$ 表示积分图，$g(x',y')$ 表示原图像。

之后利用积分图可以快速计算某区域的像素和值，如图 2.12 所示。

可以由式（2.4.3）计算，

$$D = f_1 + f_4 - f_2 - f_3 \tag{2.4.3}$$

其中 f_i 为对应点处积分图的值。

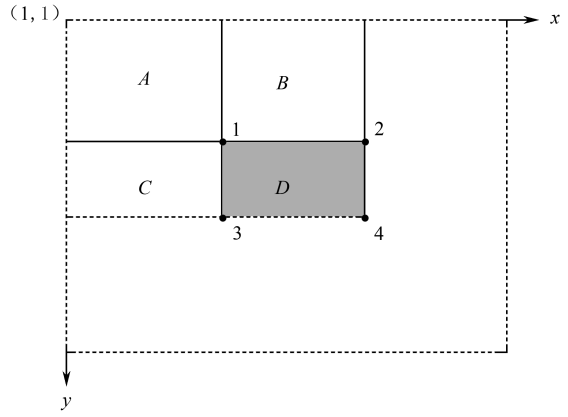


图 2.12　计算 *D* 区域的像素和值

2.4.3　Gentle AdaBoost 算法

在 AdaBoost 算法中，有强分类器、弱分类器的区别。弱分类器指在分类过程中，分类正确率略大于 50%的分类器。而将弱分类器通过数学手段组合成高正确率分类器的过程，称为训练，组合而成的拥有强分类能力的分类器，称为强分类器。Gentle AdaBoost[11]在 AdaBoost 算法基础之上，添加了自适应牛顿法，拟合加性 logistic 模型，置信度提高，提升了学习的稳定性。下面详细叙述 Gentle AdaBoost 的训练过程。

1．弱分类器训练

训练弱分类器首先需要确定 haar 特征的总数，前文已经确定共有 78460 个 haar 特征，每一个 haar 特征即可作为一个弱分类器。之后需要确定训练样本的数量，一般情况下，将正样本和负样本的数量保持一样，训练中，正负样本各使用 10000 个，共 20000 个样本进行训练。为了表述简洁，下面将 haar 特征数量记作 *N*，训练样本数量记作 *M*，其中 *N*=78460，*M*=20000。具体训练过程如下：

（1）对于 *M* 个训练样本以及 *N* 个 haar 特征，可以通过计算获得特征值二维矩阵 feature[*i*][*j*]，其中 $1 \leqslant i \leqslant N, 1 \leqslant j \leqslant M$，每一个 feature[*i*][*j*]第 *j* 个训练的第

i 个特征值，feature[i][]即为所有样本对应的某一个特征值的集合。

（2）开始循环

① value[]=feature[i][]。

② 将 value 数组从小到大进行排序。

③ For j=1:M。

$$\text{leftvalue}=\sum_{k=1}^{j}(w_k\times y_k)/\sum_{k=1}^{j}w_k \tag{2.4.4}$$

$$\text{rightvalue}=\sum_{k=j+1}^{M}(w_k\times y_k)/\sum_{k=j+1}^{M}w_k \tag{2.4.5}$$

其中，w_k 为第 k 个样本的权重，权重值在后文的强分类器训练中会计算。y_k 为样本的类型值，若 k 样本为正样本，则 $y_k=1$，否则 $y_k=-1$。leftvalue 表示前 j 个样本的集中度，rightvalue 为后 $M-j$ 个样本的集中度。

$$\text{lefterror}=\sum_{k=1}^{j}w_k\times(y_k-\text{leftvalue})^2 \tag{2.4.6}$$

$$\text{righterror}=\sum_{k=j+1}^{M}w_k\times(y_k-\text{rightvalue})^2 \tag{2.4.7}$$

其中，lefterror 为前 j 个样本的离散度，righterror 为后 $M-j$ 个样本的离散度。

If（ lefterror+righterror<fault ）

fault= lefterror+righterror ，

θ=value[j]，

$\alpha_1=\text{leftvalue},\alpha_2=\text{rightvalue}$.

End

其中，fault 表示均方误差值。

（3）保存均方值误差 fault 最小的 haar 特征的相关参数，包括左上角顶点坐标位置，haar 特征长宽值以及 fault 值及其对应的 $j,\theta,\alpha_1,\alpha_2$ 值。

经过以上训练过程，让所有的 haar 特征识别率达到了最高，并得到达到最高识别率对应的参数，可以使用这些参数对图像进行初步的检测，得到各个弱分类器对所有训练样本的检测结果。

2．强分类器训练

强分类器的训练过程即将每轮选出的拥有最低加权错误率的弱分类选出，加以组合，得到一个具有高分类能力的分类器过程，其详细过程如下。

（1）设置最低检测率为 $d_{\min}$，最大虚警率为 $f_{\max}$。准备训练样本集，其中正

样本集数量为 numPos，负样本数量为 numNeg。在本书中，numPos=numNeg=10000。

（2）初始化所有样本的权重，一般可以如式（2.4.8）设定初始权重。

$$w_j = 1/M \tag{2.4.8}$$

其中，M = numPos+numNeg，w_j 为第 j 个样本的权重，j=1,2,3,⋯,M。

（3）定义 dpre 为强分类器当前的检测率，fpre 表示当前强分类器的虚警率，t 为当前强分类中所包含的弱分类器的数量，初始值为 0。

① 计算第 i 个弱分类的加权错误率 ε_i，其中 $\varepsilon_i = \sum_{j=1}^{M} w_{t,j} \times (y_j - h_i(x_j))^2$，$h_i(x_j)$ 表示第 i 个弱分类对第 j 个训练样本的分类结果，若分类为正样本 $h_i(x_j)$=1，否则为 $h_i(x_j) = -1$。

② 计算出所有弱分类器对应的 ε_i，挑选本轮最低加权错误率的弱分类器，得到 ε_t，$\varepsilon_t = \min(\varepsilon_1, \varepsilon_2, \varepsilon_3, \cdots, \varepsilon_N)$，并且得到新的弱分类器后，$t$=$t$+1。

③ 设定当前强分类器阈值保证 $\text{dpre} > d_{\min}$。

④ 按照最优弱分类的分类结果，调整各个样本的权重，$w_{t+1,j} = w_{t,j} \cdot \exp(-y_j \cdot h_t(x_j))$。

⑤ 更新强分类器的所有参数。

（4）保存所有强分类器的参数。

3．级联分类器结构

本节中，提出并实现了递进复杂度的级联分类器结构。AdaBoost 算法中，级联分类器结构[12]的提出，大大提升了算法效率，使算法实现了实时检测的可能。级联分类器结构示意图如图 2.13 所示。

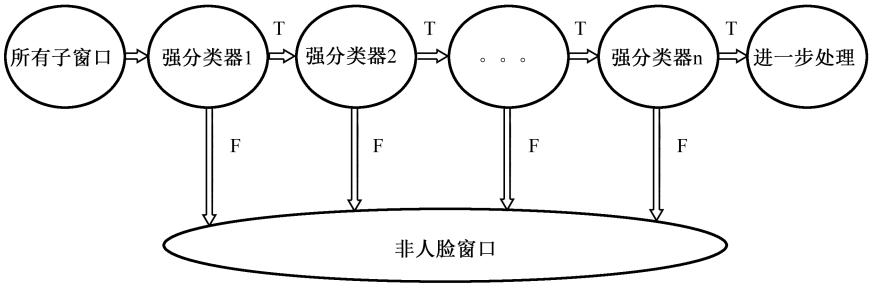


图 2.13　级联分类器结构示意图

级联分类器，其实就是利用多个强分类构造出来的。从图中可以看出，若要被识别为正样本，要求所有级的强分类器均识别目标窗口为正样本才能通过，有一级强分类识别子窗口为非人脸，则会将这个窗口拒绝。级联分类器提升检测效率的主要原因是，前级的弱分类器数量少、结构简单，可以快速过滤大量较好排除的非人脸窗口，之后通过增加级联分类器的级数，可以不断过滤难以排除非人脸子窗口，最后达到检测出人脸子窗口的目的。由于级联分类器的特性，我们需要保证每一级尽量不能将人脸误检测为非人脸，因为一旦误检测，后面的分类器也无法补救。相反的，若非人脸误检测为人脸，则可以通过后面级数的强分类器过滤掉多余的非人脸。

为了提升检测效率，本章中使用了递进复杂度的级联分类器。递进复杂度，是指每一级的强分类器，包含的弱分类数量递增。在一般情况下，弱分类器数量越大，可以达到的虚警率越低，过滤非人脸的能力也越强，但检测效率也随之下降，因此采用逐步降低最大虚警率的方法，实现递减复杂度的级联分类器结构。使用弱分类较少的强分类器作为前级，可以过滤掉大量的很好排除非人脸窗口，之后弱分类器数量较多的强分类器，进一步过滤难以排除的非人脸图像。为了保证后级的强分类能够过滤掉前级无法过滤的非人脸，需要通过训练样本的替换来实现，级联分类器的具体训练流程如下。

（1）从样本库中选取出正样本 numPos 个，负样本 numNeg 个，挑选出的样本称为训练样本，本章中，numPos=numNeg=10000，被挑选为训练样本的图像，从样本库中去除。

（2）使用 Gentle AdaBoost 算法训练，得到第一级强分类器，第一级需要结构简单的强分类器，因此本级的最大虚警率 $f_{\max}$ 可以适当提高，本章中，第一级 $f_{\max}$=70%，同时，保证高的检测率 $d_{\min}$，本章中，检测率设定为 99.99%。

（3）While（1）

① 将前一级错误分类的正样本和正确分类的负样本剔除训练样本，设被剔除的正样本数量为 Pnum，被剔除的负样本数量为 Nnum，其余保留。

② 使用已训练好的级联分类器对样本库进行检测，挑选出能被当前级联分类正确分类的正样本数量为 Pnum 个，错误分类的负样本数量为 Nnum 个，补充到训练样本库，保证训练样本数量不变，同时，成为训练样本的图像从样本库中去除。

③ 使用新得到的训练样本库进行强分类器训练。

④ 若级联分类器达到预期的检测率，训练结束。

⑤ 若级联分类器为达到预期的检测率，降低最大虚警率 $f_{\max}$，继续训练。

End

2.5　实时人脸跟踪

人脸检测的主要目的是找到人脸位置，但是光靠人脸检测无法实现将视频流中的人脸图像保存下来的目的，因此我们需要配合跟踪算法[13,14]来实现对人脸的实时跟踪，并在人脸离开视频可拍摄范围前，保存其一张相对清晰的人脸图像作为检测目标的人脸结果。如前文中所述，人脸检测的过程中，人脸会由于姿态、饰物、光线甚至直接被遮挡等原因，导致人脸检测出现短暂的检测失踪，跟踪系统也会出现误跟踪，这些都会导致跟踪结果完全错误。因此人脸检测以及跟踪算法的结合，主要有以下 3 个难点需要解决：

（1）目标出现：何时需要跟踪一个新目标。

（2）目标消失：何时需要移除一个已经跟踪的目标。

（3）目标修正：什么时候跟踪系统的跟踪结果出现错误，如何修正错误。

这些问题的出现会因为检测系统以及跟踪系统的性能不同而改变。本章中使用 Gentle AdaBoost 算法作为检测算法，粒子滤波器作为跟踪算法组合，并提出人脸检测修正策略（Face Detection Fixing Strategy）实现实时人脸检测和跟踪。

粒子滤波器[15]是跟踪算法中十分流行的算法之一，其原理是通过寻找一组在状态空间中传播的随机样本来近似地表示概率密度函数，用样本均值代替积分运算，进而获得系统状态的最小方差估计的过程，这些样本被形象地称为“粒子”，故而称为粒子滤波。在图像跟踪中，粒子滤波器的粒子就是跟踪区域中随机选取的像素点，其粒子信息一般是像素的色度，亮度等色彩信息。当视频帧流动时，帧与帧之间的信息是连续的，根据颜色的相似度可以计算出粒子的权重值，以权重值最高的粒子估算目标的位置。但是传统的粒子滤波器存在几个问题。第一，目标位置以最高权重粒子来估计并不准确，原因是最高权重粒子可能本身就不止一个，同时最高权重的粒子可能由于其他误差产生。第二，粒子滤波器的粒子随着视频帧的流动，相似度越来越低，最后会出现样本不足现象，失去跟踪能力，因此本章提出均值权重粒子滤波器算法。

2.5.1 均值权重粒子滤波器

粒子滤波器是通过从后验概率中抽取的随机状态粒子来表达其分布，是一种顺序重要性采样法。在视频中，若目标出现的时刻为 T=1，则估计 T=t 时刻的目标状态 $p(\alpha_t \mid z_{1:t})$，其中 α_t 表示目标在 t 时刻的状态，$z_{1:t}$ 表示时刻 1 至时刻 t 的所有的粒子的观测结果。这是一个贝叶斯估算模型，可以将这个过程分类一个两步递归运算：

$$p(\alpha_t \mid z_{1:t-1}) = \int p(\alpha_t \mid \alpha_{t-1})\, p(\alpha_{t-1} \mid z_{1:t-1}) \mathrm{d}\alpha_{t-1} \tag{2.5.1}$$

$$p(\alpha_t \mid z_{1:t}) = \frac{p(z_t \mid \alpha_{t-1})\, p(\alpha_t \mid z_{1:t-1})}{p(z_t \mid z_{1:t-1})} \tag{2.5.2}$$

其中 α_t 为目标在 t 时刻的状态，$z_{1:t-1}$ 为时刻 1 至时刻 t-1 时刻的所有粒子的观测结果，z_t 为 t 时刻的粒子信息，$p(\alpha_t \mid z_{1:t-1})$ 为以 $z_{1:t-1}$ 粒子的观测结果推测出的 t 时刻的目标状态的概率，其余符号的含义以此类推。式（2.5.1）为预测步骤，因为 $p(\alpha_t \mid z_{1:t-1})$ 的结果是由 1 至 t-1 时刻的粒子的观测结果推测出来的。式（2.5.2）为计算步骤，因为 $p(\alpha_t \mid z_{1:t})$ 是以时刻 1 到时刻 t 的所有粒子的观测结果进行计算，其中时刻 t 的粒子相似度信息是已知的。

设共散布了 N_s 个粒子，则在 t-1 时刻的粒子权重 $\{w_{t-1}^n, \alpha_{t-1}^n\}_{n=1}^{N_s}$ 可以根据 $p(\alpha_t \mid z_{1:t-1})$ 的概率来计算，其 α_{t-1}^n 表示第 n 个粒子的状态，w_{t-1}^n 表示第 n 个粒子的权重。但是由于实际计算中 $p(\alpha_t \mid \alpha_{t-1})$ 难以进行计算，因此一般以 $q(\alpha_t \mid \alpha_{t-1}^n)$ 来替代，其中 $q(\alpha_t \mid \alpha_{t-1}^n)$ 称为重要密度函数。这样可以得到 t 时刻各个粒子的权重：

$$w_t^n \propto w_{t-1}^n \frac{p(z_t \mid \alpha_t^n)\, p(\alpha_t^n \mid \alpha_{t-1}^n)}{q(\alpha_t^n \mid \alpha_{t-1,}^n z_t)} \tag{2.5.3}$$

之后对新得到的权重进行归一化，由式（2.5.4）计算：

$$w_t^m = \frac{w_t^n}{\sum_{n=1}^{N_s} w_t^n} \tag{2.5.4}$$

传统的粒子滤波器中，下一帧的目标位置直接由最高权重的粒子来确定。但

是在本章中，下一帧的目标位置由式（2.5.5）计算：

$$\text{Locate}_n = \frac{\sum_{t=1}^{N_s} L_n(\alpha_t \mid z_t) w_t^n}{N_s} \tag{2.5.5}$$

其中 $L_n(\alpha_t \mid z_t)$ 为 t 时刻第 n 个粒子的推测的位置信息，w_t^n 为对应粒子的权重，可以看出式（2.5.5）所计算的目标位置由所有粒子的位置信息并结合权重计算所得到，因此本算法成为均值权重粒子滤波器算法。以所有粒子的加权位置进行目标位置的估算，可以有效消除高权重粒子带来的误差，而关于粒子权重信息不断下降的解决方法，将在人脸检测修正策略中提出。

2.5.2　人脸检测校正策略

对于检测和跟踪系统，一般人脸检测系统和跟踪系统都是分开使用，互不干涉。但是事实上，人脸检测系统检测到人脸的位置信息，和跟踪系统跟踪的人脸位置信息，是可以结合使用的。人脸检测系统对人脸的定位更加准确，因此可以利用人脸检测系统对跟踪系统进行校正，这个过程称为人脸检测校正策略，具体过程如下。

（1）假设人脸检测系统在第 n 帧检测到人脸，检测位置为 LD_n。

（2）使用均值权重粒子滤波器，对检测到的目标进行跟踪。

（3）For j=n:m，在 j 表示目标被跟踪的帧数，m 表示目标能被跟踪的最后帧数。

① 跟踪系统跟踪目标，且目标在每一帧的位置为 LT_j。

② 检测系统检测每一帧是否有人脸，若检测到没有人脸，则返回步骤 1，若有，则记录检测到人脸的位置 LD_j。

③ 计算 LT_j 和 LD_j 的欧式距离，若距离小于 D_θ，则更新 LT_j，$\text{LT}_j = \text{LD}_j$，同时，在更新的 LT_j 区域进行粒子重采样，其中 D_θ 为自己设定的距离阈值。若距离大于 D_θ，则返回步骤②，认为有新的人脸进入，进行跟踪。

（4）在目标离开检测区域时，保存人脸截图，并释放这个跟踪目标区域。

这样就在跟踪的同时利用了检测系统对跟踪结果进行修正，并在修正的同时重新散布了粒子，避免了粒子的衰退问题。对于上述过程，可以用图 2.14 所示的流程图进行描述。

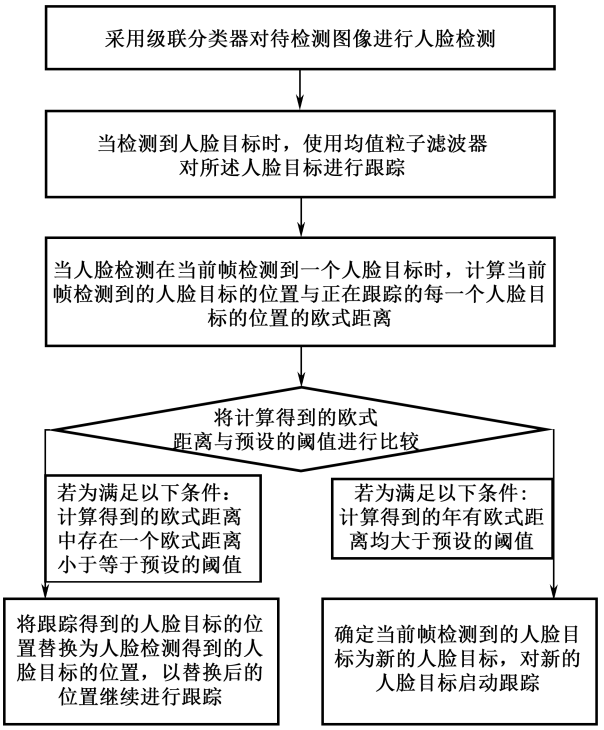


图 2.14　人脸检测校正策略流程图

2.5.3　人脸检测和跟踪实验结果分析

人脸检测和跟踪的实验中，本章使用了实时视频进行测试，视频来源是某军区大门的实时录像，共 5 段视频，共出现人脸 103 次。最后共使用级联分类器 16 级，对视频进行检测，对比了 Gentle AdaBoost 算法（以下简称 GAB）与均值权重粒子滤波器（Weighted Particle Filter，以下简称 WPF）结合策略和 Gentle AdaBoost 算法与传统粒子滤波器（PF）的结合策略的性能。表 2.1 为详细的实验结果。

其中，5 个编号的视频的具体描述如下：

1 号视频：2 人并排以正常速度进入大门。

2 号视频：2 人并排以高速进入大门。

3 号视频：所有人一列纵队进入大门。

4 号视频：单人以正常速度进入大门，每人之间间隔 3 秒左右。

5 号视频：2 人并排或者单人戴墨镜或者口罩进入大门。

表 2.1　人脸检测和跟踪实验结果

（a）GAB-PF 的检测结果

视频编号	GAB-PF 的检测结果			
	视频中出现人脸的数量	只有一张截图的人数	超过一张截图的人数	错检或者漏检的人数
1	40	10	27	3
2	22	7	12	3
3	23	10	6	7
4	12	5	3	4
5	6	1	5	0

（b）GAB-WPF 的检测结果

视频编号	GAB-WPF 的检测结果			
	视频中出现人脸的数量	只有一张截图的人数	超过一张截图的人数	错检或者漏检的人数
1	40	34	3	3
2	22	14	5	3
3	23	16	0	7
4	12	9	1	2
5	6	4	2	0

为了衡量检测和跟踪系统的性能，定义两个参数，模糊正确率和模糊错误率。

模糊正确率=被保存了至少一张截图的人脸数量/测试视频中的人脸总数量

模糊错误率=（非人脸被错误保存的数量+截图超过一张的人脸）/

测试视频中的非人脸总数量

这两个参数，类似于检测系统中的检测率和虚警率，随着模糊正确率的提升，容易将非人脸检测为人脸，同时也会增大一个人脸出现多次截图的可能，因此模糊错误率也会提升。反之，模糊正确率下降，模糊错误率也会下降。在系统中，

我们希望模糊错误率尽量低，这样保存的图像冗余信息就少，但是这样就意味着人脸漏检的可能增大，因此需要平衡这两个参数。文中所使用视频大小均为720×576 分辨率。检测框初始大小设定为 30×30 大小，检测窗每次移动窗口大小的 1/5 边长，每次对检测框的放大倍率设定为 1.2，对视频帧进行检测。由于图像的连续性，测试视频的帧速率为 30 帧/秒，因此对每一帧都进行检测意义不大，本章中，对此进行了设定，检测频率为 4 帧/次，这样即不会影响检测效果，还能提升检测效率。图 2.15 为对应视频 3 中所制作的模糊正确率和模糊错误率对应的 ROC 曲线图。

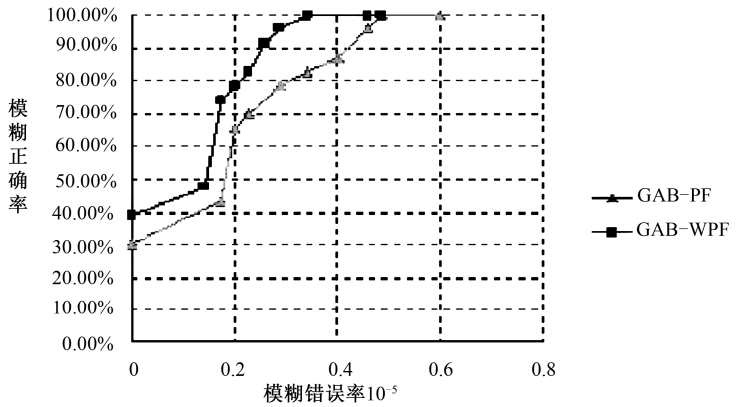


图 2.15　视频 3 对应 ROC 曲线图

由于视频中，非人脸数量远远超过人脸数量，按照 30×30 的检测窗口初始大小计算，每一帧的非人脸数量约为 35000 左右，因此模糊错误率的数量级较低。从图 2.15 可以看出，GAB-WPF 算法的曲线在各个参数点都在 GAB-PF 之上，性能较传统的 GAB-PF 算法提升较为明显。同时，表 2.1 中，进一步可以看出，使用 GAB-WPF 算法后，一张人脸对应一张截图的数量大大增加，这也说明 WPF 的跟踪算法丢失跟踪次数较少，不会导致系统多次对同一个目标的重新启动跟踪，大大降低了系统最后保存信息的冗余度。

图 2.16 为测试视频的一些截图和最终检测人脸的保存截图。从图中可以看出，即使是对戴口罩的人脸或者戴有帽子的人脸，系统依然可以正常检测、跟踪，并保存人脸截图，证明了 GAB 在人脸检测算法中具有足够的鲁棒性，同时均值权重粒子滤波器和 GAB 的协同效果也很不错。

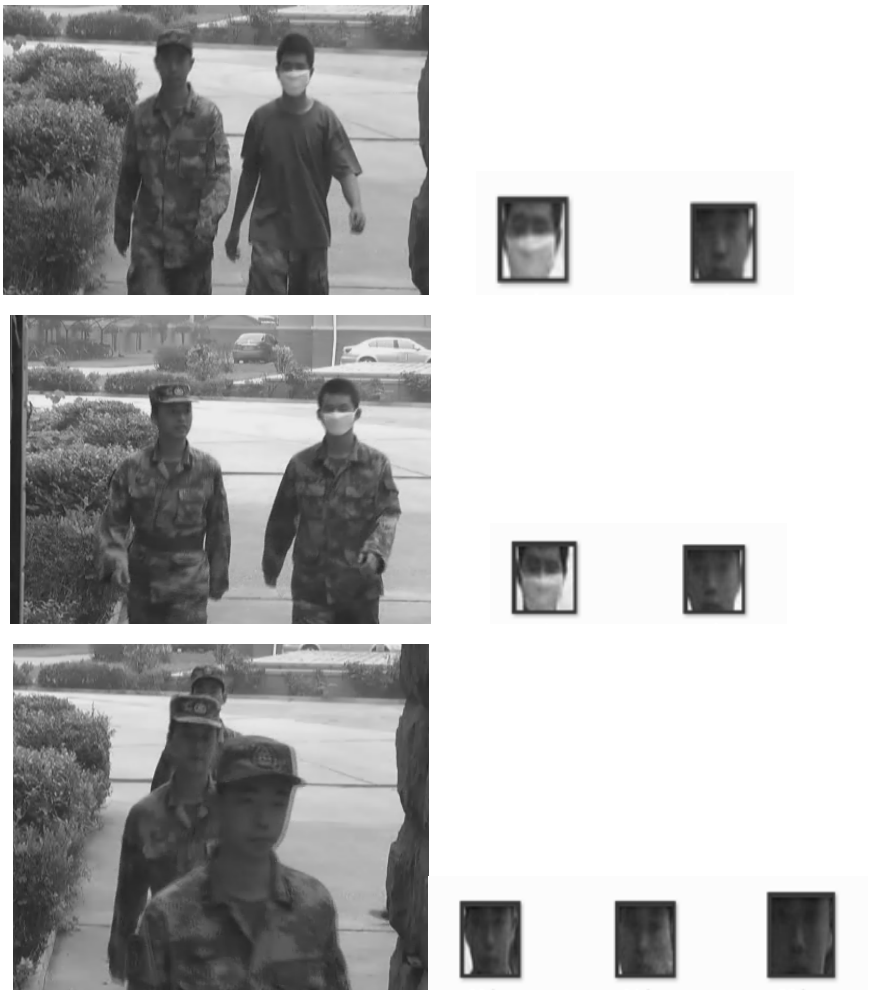


图 2.16　视频截图以及人脸检测截图保存结果示意图

2.6　本章小结

本章提出了基于人脸纹理特征点 3D 人脸配准算法。由于 3D 人脸的特殊性，配准方式和 2D 人脸有着较大不同，通过对 3D 人脸投射到 XOY 平面，获得正面人脸纹理图，将 3D 人脸特征点的检测转化到 2D 空间，使用 VOSM 算法进行特征点检测，之后再反投射到 3D 人脸得到 3D 人脸特征点的位置。最后以特征

点为基准点，建立新坐标系，并重新构造三角网格，实现了 3D 人脸数据库的重建以及配准。后文中，虽没有直接使用到本章的 3D 配准方法，但是后文中的所有 2D 图像的配准算法都是本章配准方法的基础，因此本章配准算法的实现，为后文的研究打下了良好的基础。

关于基于均值权重粒子滤波器的人脸检测跟踪算法，首先简要介绍了常用的人脸检测算法，详细介绍了 Gentle AdaBoost 算法的训练过程，提出了递进结构的级联分类器，提升检测效率，达到实时检测的目的。之后提出了均值权重的粒子滤波器算法，解决了高权重的粒子问题，实现跟踪系统，并提出人脸检测校正策略，结合了人脸检测和跟踪系统的位置信息，提高了系统对于人脸跟踪的准确度，并解决粒子滤波器样本衰退问题。从实验结果可以看出，相较于传统的 GAB-PF 算法，GAB-WPF 算法有效提升了跟踪准确度，并减少了保存图像的冗余信息，性能提升明显。

本章参考文献

[1] GuptaS, Markey M K, Bovik A C. Advances and Challenges in 3D and 2D+3D Human Face Recognition [M]. Columb us F. Pattern Recognition Theory and Application, New York: Nova Science Publishers, 2008, 63-103.

[2] Federico M. Sukno, Sebastian Ordas, Constantine Butakoff, et al. Active Shape Models with Invariant Optimal Features: Applicaiont to Facial Analysis[J]. IEEE Transactons On Pattern Analysis And Machine Intelligence, 2007, 29(7):1105-1117.

[3] Wang Y, Liu J, Tang X. Robust 3D face recognition by local shape differenceboosting[J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2010, 32(10): 1858-1870.

[4] Ben Amor, B., Srivastava, A., Daoudi, M, etc. 3D Face Recognition under Expressions, Occlusions, and Pose Variations [J]. IEEE Transactions on PAMI, 2013, 35(9): 2270-2283.

[5] Queirolo C C, Silva L, Bellon O R P, et al. 3D face recognition using simulated annealing and the surface interpenetration measure [J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2010, 32(2): 206-219.

[6] D. Ramanan, D. Forsyth. Using temporal coherence to build models of animals. In Proc. Int. Conf. Comp. Vision(ICCV)[C]. Nice, France, 2011, 338-346.

[7] http://www.cvchina.info/2011/05/21/vosm-library/.

[8] Jian Huang, Ziling Su, Ruomei Wang. 3D Face Reconstruction Based on Improved candide-3 model. Digital Home (ICDH), 2012 Fourth International Conference on Digital Object[C]. Guangzhou: IEEE, 2012, 438 -442.

[9] T. Cootes, G. Edwards and C. Taylor. Active Appearance Models[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, vol.23:681-685.

[10] Belhumeur P.N.，Hespanha J. P.，Kriegman D. J. Eigenfaces vs. Fisherface: Recognition Using Class Specific Linear Projection[J]. IEEE Trans. PAMY, 1997, 19(7), 711-720, July.

[11] 王建. 基于 GentleAdaboost 算法的人脸检测研究[D]. 成都：电子科技大学，2011.

[12] Viola, P., Jones, M.J.. Rapid object detection using a boosted cascade of simple features. IEEE Computer Society Conference on Computer Vision and Pattern Recognition[C]: IEEE, 2001, 1, 511-518.

[13] Tu JL, Tao H, Huang T. Face as mouse through visual face tracking[J]. Computer Vision And Image Understanding, 2010, 108(1):35-40.

[14] Gerard Medioni, Isaac Cohen, et al. Event detection and analysis from video stream[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, 23(8):873-875.

[15] 夏利民, 张良春. 基于自适应粒子滤波器的物体跟踪[J]. 中国图像图形学报, 2009, 14(1): 112～117.

第 3 章

人脸验证和素描人脸识别

本章首先提出了基于 SIFT 特征的人脸验证算法。传统人脸验证，大都需要有监督训练的机器学习过程，利用 SIFT 特征对两幅图像进行匹配，得到匹配点后，将匹配点划分为数量特征和位置特征，结合使用，实现了无须训练的人脸验证。实验证明该算法在验证率基本没有变化的情况下，大大提升了算法的适用范围。

另外，还提出了结合分块 LBP 特征的素描人脸识别算法。传统的同质人脸识别已经比较成熟，本章研究了异质人脸识别的一个热门分支，素描人脸识别。这里算法是利用 LBP 算子，寻找素描人脸图像和光学照人脸图像的纹理相似处，并利用 DOG 滤波器增强两个图像的纹理信息。之后对图像进行分块处理，利用机器学习的思想，寻找能有效进行素描人脸识别的特征块，并赋予权重。最后计算加权总距离，来判断图像间的相似度，实现了素描人脸识别。

3.1 人脸验证简介

人脸验证，是人脸识别领域的一个分支。人脸识别包括两大类：①人脸身份识别，根据人脸图像识别人物的身份，解决是谁的问题，是个一对多的匹配过程，这也是前一章研究的主要内容；②人脸验证，判断图像中的人脸是否是指定的人，解决是不是某人的问题，是一对一的匹配过程，广泛应用在安防领域中，一直都

是国内外研究的热点和重点，已经有不少研究成果。在国外，Belhumeur 等人使用了 Fisher 特征人脸算法，实现人脸识别，之后 Kittler[1-3]等人在 LDA 的基础上引进了客户相关子空间（client specific）的验证方法，实现人脸验证。在国内，在上述研究基础上，主要有陈伏兵等人对 2DPCA 推广提出了模块化 2DPCA 方法，之后袁宁[5]等提出了 2DPCA 和 CSLDA 改进算法。这些算法都属于线性分类算法，也是目前常用的验证算法类型[6,7]。

现有的人脸验证算法，大多还是使用 PCA 降维，获得人脸特征，再使用 LDA 获取类间差值，来确定人脸是否匹配，属于有监督训练过程的机器学习算法。近年，在人脸验证中，比较有代表的成果是 Hae Jong Seo 等[8]以及 Juan Ramón Troncoso-Pastoriza 等[9]提出的算法。Hae Jong Seo 在 2011 年提出局部自适应回归核的人脸验证算法，该算法很好地解决了多表情下的人脸验证，在各个人脸数据库中的测试有良好的结果。但是这个算法依旧是需要训练，验证时，需要已经训练过的数据库的支持。2013 年，Juan Ramón Troncoso-Pastoriza 等提出非交互的人脸验证算法，文中指出现阶段存在的人脸验证算法基本都限于和服务器端已存的人脸图像进行对比，无法对外部没有训练的图像对比。本章提出使用 Gabor 变换结合 SVM 进行训练，获得分类器，最后使用不同的人脸库进行验证，获得了较好的效果。这个算法解决了传统人脸验证中需要训练才能验证的问题，但是需要计算 Gabor 特征，并且分类器维度较高，验证效率较低，不利于实时验证。

基于上述情况，这里提出采用尺度不变特征（Scale-Invariant Feature Transform，SIFT）算法来确定图像是否匹配。SIFT 算法在 1999 年由 Lowe 首次提出，并在 2004 年作为一种特征应用于图像匹配领域。SIFT 特征具有尺度不变性，可在图像中检测出关键点，是一种局部特征描述算子，可以有效地寻找到两幅图像间存在的相似的特征点数。本章提出的算法，和传统人脸验证算法区别的是，算法是试图寻找两幅待验证图片的相似处，没有机器学习过程，没有监督训练过程，对样本库外的图像有更好的兼容性。但是人脸差异性不是非常明显，仅以 SIFT 作为特征来匹配两幅图像[10]，还不足够。为了解决上述问题，这里特提出使用 SIFT 特征基础上引入了分块特征，对各个分块进行 SIFT 匹配，将所有分块的 SIFT 匹配点组合成一个向量，计算两幅图像间的匹配向量距离，比较匹配向量的相似度，最后判断两幅图像是否匹配。算法使用中科院 CAS 数据库以及 FERET 数据库进行了测试，验证了算法的有效性。

3.2 SIFT 匹配算法

3.2.1 SIFT 算子

SIFT 算法在 1999 年由 Lowe 首次提出，并在改进后于 2004 年正式提出了一种基于尺度空间的、对图像缩放、旋转甚至仿射变换保持不变性的图像局部特征描述算子——SIFT 算子[11]，也称为 SIFT 特征。下面简要介绍一下 SIFT 特征的提取过程。

（1）将图像转化到高斯差分空间。

若输入的二维图像为 $I(x,y)$，则其对应的高斯尺度图像 $L(x,y,d)$为：

$$L(x,y,d)=G(x,y,d)\times I(x,y) \tag{3.2.1}$$

其中 $G(x,y,d)$为高斯核函数，d 为尺度空间因子。之后利用高斯尺度图像构建高斯差分尺度图像，如式（3.2.2）所示：

$$D(x,y,d)=(G(x,y,d)-G(x,y,kd))\times I(x,y)=L(x,y,d)-L(x,y,kd) \tag{3.2.2}$$

其中 k 为相邻层图像的平滑因子。

（2）尺度空间极值点检测，获得尺度不变性。

对每个像素点在其图像空间和 DOG 尺度空间的邻域中搜索极值点，初步得到特征点的位置。以某像素点为中心，将同 DOG 尺度空间的 8 个相邻点，以及上下相邻尺度对应的 9×2 个点，共 26 个点构成的邻域中进行比较，以确保在尺度空间和二维图像空间都能检测到极值点。这些极值点将作为粗特征点，之后利用 DOG 函数的二阶 Taylor 展开式进行特征点精确定位。

（3）为特征点分配方向值。

首先计算高斯空间特征点的梯度模和方向，如式（3.2.3）与式（3.2.4）所示：

$$m(x,y)=[(L(x+1,y)-L(x-1),y)^2+(L(x,y+1)-L(x,y-1))^2]^{\frac{1}{2}} \tag{3.2.3}$$

$$\theta(x,y)=\tan^{-1}\left(\frac{(L(x,y+1)-(L(x,y-1)}{(L(x+1,y)-(L(x-1,y)}\right) \tag{3.2.4}$$

其中 $m(x,y)$为特征点梯度幅度，$\theta(x,y)$为特征点方向，然后使用梯度直方图确定特征点主方向和其辅助方向，实际计算中，梯度直方图的范围是 0～360°，其中每 45°一个柱，总共 8 个柱。

（4）获取 SIFT 特征向量。

首先将特征点作为中心，将坐标轴 x 轴旋转到特征点主方向上，之后以特征点为中心，取 16×16 邻域的像素块，作为采样区域。把这个 16×16 的像素区域以 4×4 大小的像素块划分子块，可以得到 16 个子块，计算每一个区域中像素在 8 个方向的梯度直方图，则一个区域可以得到一个 8 维向量，最后将 16 个区域 8 维向量组合，得到一个 128 维的向量，这个向量即为 SIFT 的特征向量，即 SIFT 算子，如图 3.1 所示。

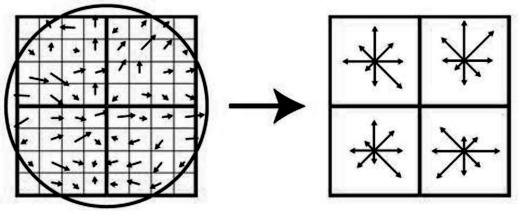
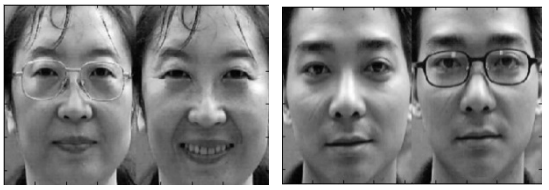


图 3.1　8×8 区域 SIFT 向量计算

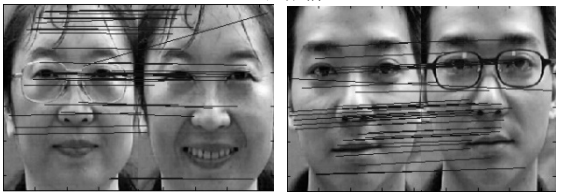
图 3.1 中只列出了其中一个 8×8 大小的区域，因此最后得到是 4 个子块的 4×8=32 维的向量。还有 3 个 8×8 个大小的区域，其计算方式和图 3.1 相同，每个区域可以计算得到一个 32 维的向量，组合后将得到 4×32=128 维向量，作为 SIFT 的特征向量。

3.2.2　SIFT 匹配

如 3.2.1 节中所述，得到图像的特征点以及对应的特征向量后，我们采用关键点特征向量的欧式距离来作为两幅图像中关键点的相似性判定度量。为了排除因为图像遮挡和背景混乱而产生的无匹配关系的关键点，采用比较最近邻距离与次近邻距离的方法，比如两幅需要匹配的图像，第一幅称为图像 1，第二幅称为图像 2，取图像 1 中的某个关键点，并找出其与图像 2 中欧式距离最近的前两个关键点，最近的两个关键点距离为 d_1，次近的两个关键点距离为 d_2，ratio=d_1/d_2，其中当距离比率 ratio 小于某个阈值的认为是正确匹配。对于错误匹配点，由于特征空间的高维性，相似的距离可能有大量其他的错误匹配，从而它的 ratio 值比较高。降低 ratio 值会减少匹配点数量，但是找出的匹配点会更加稳定。图 3.2 是一些 SIFT 算法的匹配效果较好的一些图像。



（a）匹配前

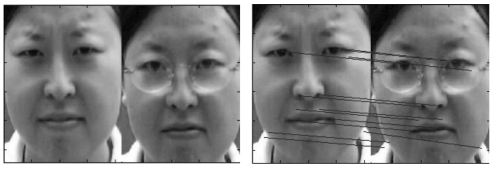


（b）SIFT 算法匹配后

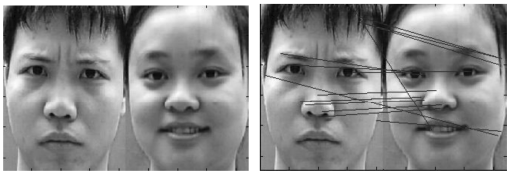
图 3.2　SIFT 算法匹配

3.2.3　SIFT 数量特征匹配分析

由 3.2.2 中图 3.2 可以看出，SIFT 匹配算法应用在人脸验证中，是有效的，这种使用匹配点数量来进行配对的特征，称为数量特征。但是，可以看出在图 3.2（b）第一对图像中，有一对错误的匹配点，事实上，SIFT 特征在许多图像对匹配中，只用数量特征，匹配效果并不理想，不仅有错误匹配点，还有匹配点数量不足的情况。因此 SIFT 应用在人脸验证中，不仅需要匹配点相对位置一致，还需要有足够的匹配点数，这样才能认为这一对图像是同一个人。图 3.3 是 SIFT 算法难以匹配的图像示例。



（a）



（b）

图 3.3　SIFT 匹配失败图示

从图 3.3 可以看出，图 3.3（a）为同一人的两幅图像的匹配结果，图 3.3（b）为不同两人的两幅图像的匹配结果。从匹配点数量上来看，两对图像结果一样，因此以匹配点数量作为人脸验证的判断标准，是无法区分这两种情况的。但是从图像上可以看到，图 3.3（b）中两幅图像的许多匹配点位置匹配是错误的，如果以欧式距离来计算匹配点间的距离，有许多匹配点的欧式距离远大于合理的距离。因此不能只用数量特征作为 SIFT 特征进行人脸验证，需要配合其他类型的特征[12]再进行验证。

3.3　SIFT 位置特征的人脸验证算法

本节中，提出了结合 SIFT 位置特征的人脸验证算法。3.2 节中，已经证明 SIFT 算法在人脸验证中有一定效果，但是只利用 SIFT 匹配点数量作为验证标准，效果还不够理想。分析实验结果可以知道，SIFT 算法对图像匹配过程中有两种特征可以利用，一是匹配点数量，匹配点数量足够多的图像一般都是匹配图像；二是匹配点的相对位置。实验中证明，只用匹配点数量作为特征验证效果不理想，主要原因是，有许多匹配图像对和非匹配图像对的匹配点数量十分接近，导致无论如何划分匹配点阈值，也不能很好地区分图像是否匹配，因此要利用匹配点间的相对距离作为特征，进行验证。由图 3.3 可知，非匹配图像和匹配图像的匹配点数量近似的情况下，有许多匹配点相对距离非常大。本章提出对图像分块[13]，计算各个分块中的特征点数量，并利用每一块中的特征点数量组成特征向量，最后计算向量相似度来确定是否匹配。具体方法如下。

（1）对于需要验证的一对图像，第一幅称为 Img1，第二幅称为 Img2，对两个图像分块，将两幅图像都划分为 $M\times N$ 个子块，两幅图像的子块分别记为 Img1_i，Img2_i，i=1,2,3,…,$M\times N$，如图 3.4 所示。

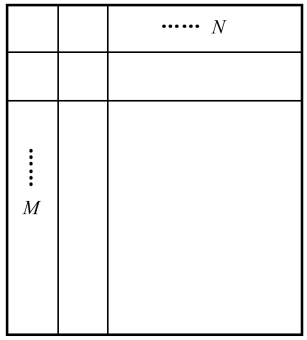


图 3.4　图像分块示意图

（2）对于两幅图像进行 SIFT 匹配，设两幅图像的匹配点数量为 K。

（3）对所有匹配点进行区域统计，统计两幅图像各个分块中的匹配点数量，设 Img1 对应的向量为 hist1，Img2 对应的向量为 hist2，向量维度均为 $M\times N$，称为匹配向量，其中 hist1_i 为 Img1_i 中对应的匹配点数量，hist2_i 为 Img2_i 中对应的匹配点数量。

（4）得到两幅图像的匹配向量后，以匹配向量匹配的方式，按式（3.3.1）判断两个向量的相似程度 s，若大于一定阈值，则认为满足匹配条件；若小于，则不满足匹配条件。

$$s=\sum_{i=1}^{M\times N}\frac{1-\text{abs}(\text{hist1}_i-\text{hist2}_i)}{\max(\text{hist1}_i,\text{hist2}_i)}\times 100\% \tag{3.3.1}$$

使用对匹配向量进行向量匹配，相比直接判断匹配点相对距离的方法，更具有统计意义，匹配结果更为可靠。相似率的阈值，可以根据环境需求进行调整。

验证过程

利用 SIFT 算法得到了一对需要验证的匹配点数量以及匹配向量，就可以对图像进行匹配验证。具体步骤如下

（1）对需要验证的一对图像 Img1、Img2 进行平均分块，如图 3.4 中所示，获得分块 Img2_i、Img2_i，其中 i=1,2,3,…,$M\times N$，其中 M 和 N 分别是纵向和横向划分的分块数量，$M\times N$ 为总分块数量。

（2）对 Img1、Img2 进行 SIFT 匹配，得到两个图像的匹配点数量 K。

（3）若匹配点数量大于阈值 x，则进行第（4）步，否则认为两幅图像不匹配。

（4）计算匹配点位于各个区域的数量，得到对应的匹配向量 hist1、hist2，其中 hist1_i、hist2_i 的值即为分块 Img1_i、Img2_i 中匹配点的数量。

（5）计算向量 hist1_i、hist2_i 的相似度，若大于阈值 θ 则认为是匹配，反之认为不匹配。

图 3.5 为该过程的流程图。

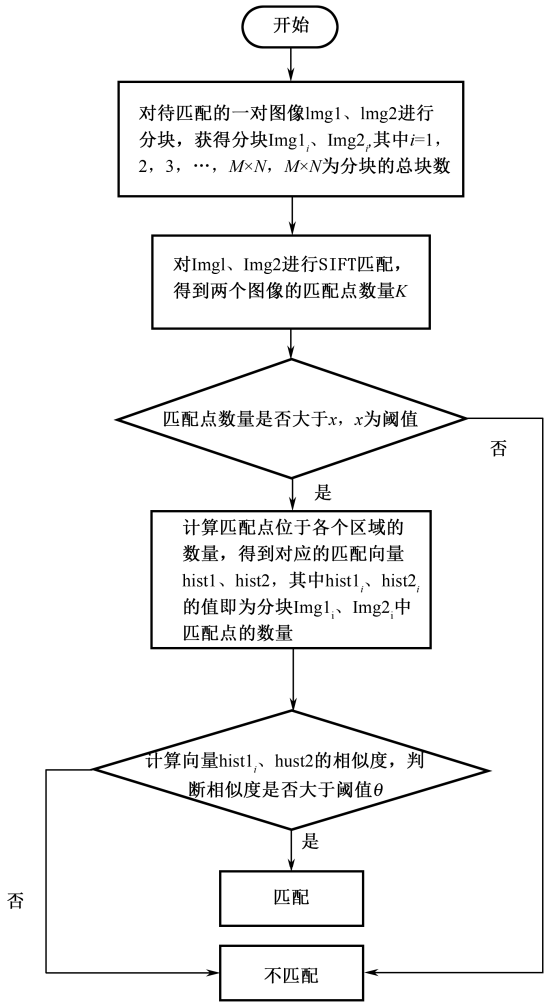


图 3.5　验证过程流程图

3.4　人脸验证实验结果与分析

本章首先使用了中科院计算技术研究所中的人脸数据进行算法有效性的验证，其中共挑选 20 人，每人有对应 3 幅图像，其中两幅戴眼镜，眼镜之间略有区别，一副不戴眼镜，3 幅图像间表情略有区别，图 3.6 是一个人的所对应的 3

幅图像例子。

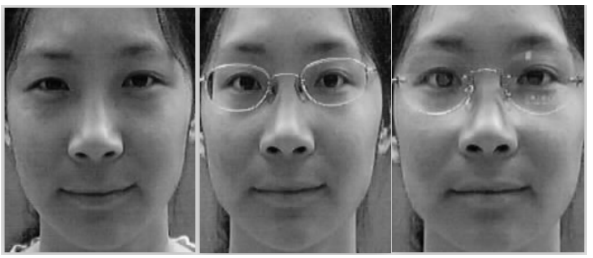


图 3.6　一个人所对应的 3 幅图像

按上述方法挑选人脸能得到 60 幅人脸，实验中有以下 2 个测试数据需要先定义：

（1）虚警率：本来是同一个人的一对图像误报为不同人的概率。在本实验数据库中，1 幅图像有 2 幅对应为同一个人的其他图像，所以匹配结果本应该同一个人的次数应为 2×60=120 次，若出现虚警次数为 x，则最终虚警率为 x/120×100%。

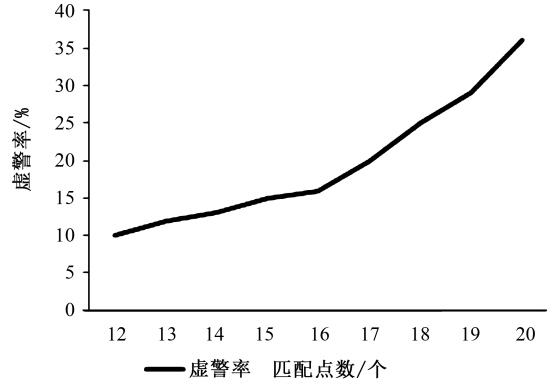
（2）漏警率：本来是不同人的一对图像，系统误认为是同一个人的概率。在本实验数据库中，1 幅图像除了对应的 2 幅同一个人的图像外，其余 57 幅都是不同人的，因此本应报警为不同人的次数为 60×57=3420 次，若本应报警却没有报警的次数为 y，则最终的漏警率为 y/3420×100%。

3.4.1　SIFT 数量特征的人脸识别

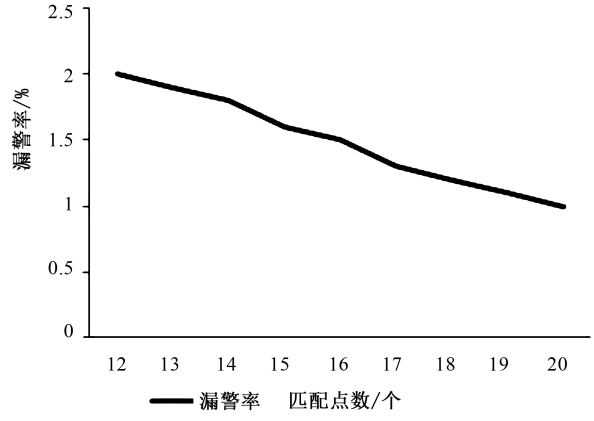
这里首先测试了只使用 SIFT 数量特征进行验证的方法，即只能使用匹配点数量作为特征进行匹配，而阈值则是使用某个匹配点数量，大于这个数量认为一对图像是匹配的，小于则是不匹配。阈值的改变会影响到虚警率和漏警率，一般来说，提高阈值会降低漏警率，但是会提升虚警率，图 3.7 是 SIFT 算法中，使用数量特征各个阈值的验证效果。

如图 3.7 所示，阈值的选取对虚警和漏警的影响是矛盾的，一般来说，我们需要在其中寻找一个相对平衡的点，在实际验证中，虚警率不能超过 20%，不然需要太多的人为干预，而漏警率不能超过 1%，因为一旦漏警，则会有重要的安全隐患。而图 3.7 中可以看出，仅用 SIFT 进行验证，无法满足上述条件，当阈值取 20 的时候，漏警率接近了 1%，但是对应的虚警率已经高达 35%以上了，而其他阈值显然和条件都差距很大，因此单独使用 SIFT 的数量特征是无法满足

验证需求的。



（a）阈值和虚警率的关系



（b）阈值和漏警率的关系

图 3.7　训练样本影响

3.4.2　结合 SIFT 位置特征的人脸验证

设定要加入位置特征验证，首先我们需要做的是使用匹配点数量筛选掉不可能是匹配图像的情况，由图 3.7 可以知道，匹配点作为阈值，当在 13～17 之间时，虚警率可以满足在 10%到 20%，在这个基础上，去掉那些匹配点位置差距较大的图像，则可以再次降低漏警率，使结果满足需求。之后的验证过程需要考

虑的有以下 2 个因素，一是分块的数量，理论上块数越多，则图像分割的越精细，效果会更好，但是由于匹配点数量有限，一旦大量分块，有许多分块中的匹配点数量都是 0，这样对匹配向量进行向量匹配，尽管相似度很高，但是却没有意义，因此需要找到合理的分块数量。本章中使用 3×3 的分块和 4×4 的分块进行实验测试。二是向量匹配的相似度阈值，相似度至少要大于 50%以上的阈值才是合理的阈值。表 3.1 是 4×4 分块中，使用不同阈值以及相似度的虚警率。

表 3.1　各个参数下虚警率和漏警率变化

（a）各个参数下的虚警率变化

匹配点数 / 相似度	13	14	15	16	17
50	8.30%	10.00%	10.83%	12.50%	15.83%
60	12.50%	15.83%	18.30%	20.83%	22.50%
70	19.17%	20.00%	21.67%	24.17%	27.50%

（b）各个参数下的漏警率变化

匹配点数 / 相似度	13	14	15	16	17
50	2.02%	1.99%	1.75%	1.52%	1.40%
60	1.46%	1.20%	0.88%	0.70%	0.41%
70	0.31%	0.29%	0.26%	0.26%	0.20%

如表 3.1 所示，在结合使用相似度计算之后，虚警率有所提高，但是还在可以接受的范围，对应的，漏警率大大降低了。从表中可以看到，在使用匹配点数为 13 至 15,相似度为 60 至 70 的阈值时，有多个结果满足需求，可以根据实际需求，选择最优的参数搭配。将相似度固定，取不同的匹配点数作为阈值将表 3.1 做成 ROC 曲线后，其结果如图 3.8 所示。

本章中最后使用的是匹配点数 13，相似度 70%的参数，保证漏警率达到最低，满足安全性能的需求。

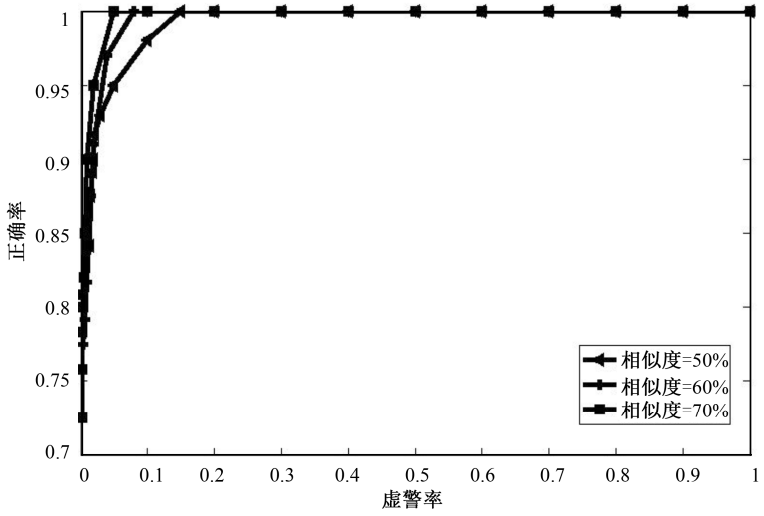


图 3.8　不同相似度下的 ROC 曲线图

3.4.3　和传统人脸验证算法的对比

传统算法中选用袁宁等提出了 2DPCA 和 CSLDA 作为对比，数据使用文献 5 中的参数，详细结果如表 3.2 所示

表 3.2　和传统人脸验证算法的对比

算法名称	参数设置	虚警率	漏警率
2DPCA+CSLDA[5]	分块 11×3　r=3	15%	2.92%
2DPCA+CSLDA[5]	分块 5×3　r=6	16.6%	2.51%
本章算法	分块 4×4	19%	0.29%

由表 3.2 可以看出，传统算法中，虚警率较新算法要更低，然而漏警率过高。前文中提过，漏警率在实际中要求尽可能为 0，安全性是首要考虑的，因此本章算法在安全性方面更为可靠。

同时，为了和其他论文中的算法进行对比，还使用本章算法对 FERET 数据库进行了测试。测试中，从数据库中随机选取 20 人，每人随机选取 3 张图像进行验证。FERET 数据库中选取的部分图像示例如图 3.9 所示。

图 3.9　FERET 图像示意图

本章算法的 ROC 曲线以及其他算法的 ROC 曲线对比图如图 3.10 所示。

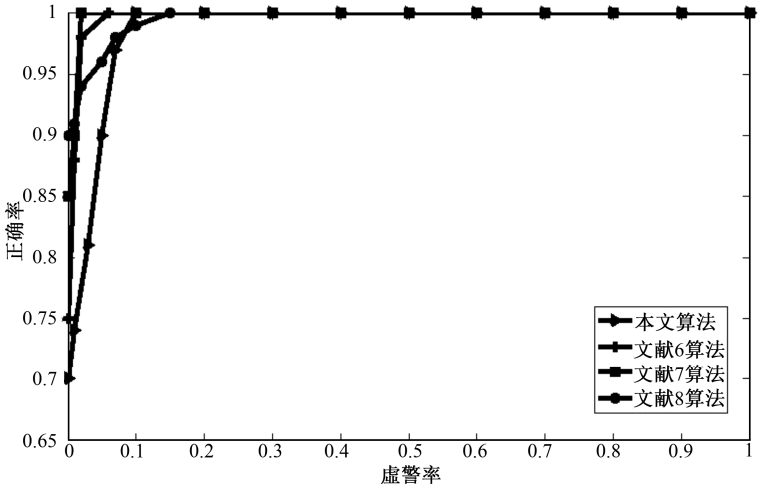


图 3.10　不同算法在 FERET 库 ROC 曲线对比图

由图 3.10 中可以看出，在 FERET 数据库中的算法取得了较好的效果，但是由于这两个算法训练的图像和验证的图像的人是来自同一个人，因此在 FERET 数据库的测试结果比本章中算法更好，是预期中的结果。本章算法和文献 9 算法在验证时，均不涉及到训练图像问题，文献 9 算法在虚警率为 0%时，取得了更高的验证正确率，但是本章算法在更低的虚警率达到了 100%的验证率，也证明了本章算法的有效性，在验证性能没有太大差异的情况下，大大提升了算法的适用范围。另外由于在验证 FERET 数据库时，挑选的样本有些表情变化较大，对本章算法也有一定的影响，因此验证效果没有在 CAS 数据库中的好。

3.5　人脸识别简介

人脸识别是指对于给定的一幅人脸作为输入，在待识别库中进行识别，寻找出和输入人脸同一个人的人脸图像。和人脸检测以及人脸验证不同，人脸识别是一对多问题，因此一般使用的方法都是使用某一类算法计算输入图像和待识别图像的相似度，之后按识别度对待识别图像排序，相似度最高的图像认为是同一人的图像。传统的人脸识别算法研究已经比较成熟，常用算法包括特征脸（Eigenface）、Gabor、PCA 结合 LDA 以及以上算法的改进算法等都在传统人脸识别，即同质人脸识别中都取得非常好的效果，但是在异质人脸识别方向，相关研究还十分缺少，因此本章主要研究了异质人脸识别其中一个应用，素描人脸识别。

异质人脸识别[14]，是人脸识别领域的一个分支，也是近年来人脸识别领域的一个新的研究热点和难点。与之前章节的研究有所不同的是，区别于一般人的人脸识别，异质人脸识别是指识别库中的图像和待识别目标图像，获取方式不同，在这个基础上进行的一种人脸识别，比如识别目标为红外图像、深度图像、素描图像等，而识别库为可见光图像等。异质人脸识别在国内外已经有一些学者进行研究，目前研究成果集中在红外人脸识别，即识别目标源为红外获取的人脸图像，识别目标库为可见光人脸图像，这种识别的应用主要在门禁系统中。素描人脸识别[15]，也是异质人脸识别的一种，是指将一个人的可见光照片和其对应的素描照进行识别，最终达到通过输入素描照寻找到对应人的可见光照片的过程，是刑侦领域的研究热点，关于这个领域，国外以 Brendan F. Klare 博士的研究比较成熟，研究成果比较多，而在国内这方面的研究还比较缺乏。

素描人脸识别分为两大类，第一类是画师在可以见到目标的可见光照片的情况下，进行素描，以这种素描照作为待识别目标对其他可见光图像进行识别的过程，称为可见素描人脸识别；第二类则是由人对目标进行脸部特征描述，由法医根据描述对目标进行素描重现，以这种方式获取的素描照作为待识别目标对其他可见光图像进行人脸识别，称为法医素描人脸识别[16]。由于数据库的限制，主要是缺乏专业法医以特征描述重现人脸的素描照数据库，本章目前只研究了第一类的素描人脸识别。在今后的研究中，将会以目前的研究成果作为研究基础，进一步探究在法医素描人脸识别中，算法的优化问题，使本章算法可以推广到实际的刑侦中使用。

和传统的人脸识别相比，素描人脸识别的重点有所改变，传统的人脸识别[1]问题主要集中在光照、姿态、饰物、表情等条件限制，影响了识别率，这主要是

因为传统的人脸识别更多的是应用在一般大众场合，不可能对识别目标有动作、穿着、表情等硬性要求，因此识别要求有更高的鲁棒性。而素描人脸识别由于应用的方面主要就是刑侦，因此，一般来说，识别库中的图像都是面部清晰，没有饰物遮挡或者特殊表情的标准证件照，同时，画师或者法医的素描照也会是正面人脸素描照，无饰物或者其他遮挡。因此素描人脸识别需要解决的主要是素描人脸如何与光学照片相匹配的问题，其他外部条件暂且不予考虑。而素描人脸图像和可见光人脸图像的最大区别是素描人脸没有色彩信息，面部特征清晰[17]，这一点类似于灰度图像。另外，由于素描图像的匮乏，一个人只能有一张对应的素描图像作为训练使用，因此素描人脸识别还是基于单幅图像的人脸识别算法。而单幅图像的人脸识别算法本身是人脸识别的一大难题，这也造成了传统人脸识别算法，如 PCA、GABOR 等效果不佳。

基于上述情况，提出采用 LBP[18]算子来刻画素描图像和可见光图像的相似性。LBP 图像的特点是可以有效刻画图像纹理[19]，同时放弃颜色信息，符合素描人脸照和可见光人脸照的共同特性，再结合图像分块算法，利用机器学习算法思想特征提取，寻找到能有效识别素描照和可见光照相似性的特征，并使用 CUHK 的素描人脸数据库[20]对算法进行测试，验证了算法的有效性。

3.6 LBP 识别算法

3.6.1 LBP 基本算子

LBP 算子[21-23]是对图像纹理进行描述的一种有效手段，将图像信息的颜色信息去除后，保留了图像的变化信息，如亮度和色度的变化规律。基本的 LBP 算子是一个 3×3 的矩阵，以中心点为的像素值作为阈值，其他周围 8 个点和中心点进行比较，若大于中心点，则为 1，若小于中心点则为 0。设中心点坐标为（0,0），阈值为θ，且周围 8 点位于坐标（1,1），（1,0），（1,−1），（0,−1），（−1,−1），（−1,0），（−1,1），（0,1），则有式（3.6.1）：

$$f(x,y)=\begin{cases}1 & \text{当 } f(x,y)>\theta \\ 0 & \text{当 } f(x,y)<\theta\end{cases} \tag{3.6.1}$$

这样一来，将一个 3×3 的子块以中心点周围 8 个点按照上述坐标顺序，进行排序，将组成一个 8 位二进制数，将这个二进制数转换为十进制数替代中心点

原数据，即是这个算子的 LBP 值，如图 3.11 所示。

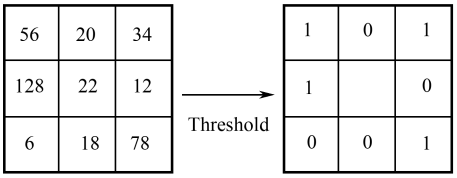


图 3.11　LBP 基本算子

如图 3.11 中，以上文给出的坐标顺序读取二进制数，则 LBP 值为 $(10100110)_2 = 166$。对于任意矩阵，将边缘部分补一行 0，一列 0，按照上述处理方法即可得到这个矩阵的 LBP 矩阵，如果矩阵为图像矩阵，则得到这个图像对应的 LBP 图像。图 3.12 为一副可见光人脸照片的灰度图像及其对应的 LBP 图像。

（a）原图像灰度图像　　（b） LBP 图像

图 3.12　原图像和 LBP 图像的对比

图像数据由 CUHK 素描人脸数据库所提供，训练数据都经过面部特征位置的配准，如眼睛、鼻子、嘴巴等关键特征，均基本在同一坐标位置，像素值差距不大于 10，图像大小均为 250×200。

3.6.2　LBP 人脸识别

LBP 识别一般以 LBP 直方图进行识别，即将 LBP 图像进行直方图统计，由 3.6.1 所述可知，以 LBP 基本算子生成的 LBP 图像，所有像素值应该是 0～255 范围之间，以这个图像做直方图统计，可以知道各个数值的分布范围。若把直方

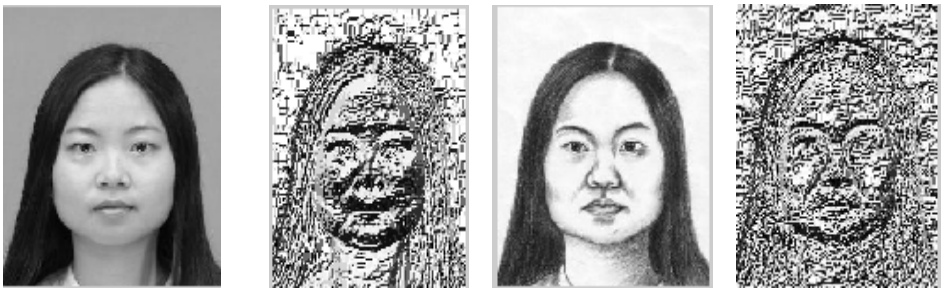
图看做一个向量，则是一个 256 维的行向量，以α_i表示。对于每一个人脸图像，都可以经过 LBP 处理得到其对应的 LBP 向量，通过计算两个图像 LBP 直方图的距离，判断两者的相似性。最初，Kullback 提出用交叉熵的方法进行 LBP 的统计测定，之后 Sokal 将这个算法称为 G 统计，也称为 Log 概率统计[23]，经过化简后，其表达式如式（3.6.2）所示：

$$L(S,M)=\sum_{b=1}^{B}S_b\log_{10}M_b \tag{3.6.2}$$

其中，（S,M）分别为 2 个样本的 LBP 向量，b=1,2,…,256，S_b和 M_b分别为某像素值出现的概率，距离 L 即反映了（S,M）向量之间的相似度，后文中，没有特意说明的情况下，LBP 距离即指式（3.6.2）中的距离。当有数个样本的情况下，距离最小的 2 个即认为是同一类样本。

3.6.3 LBP 算法分析

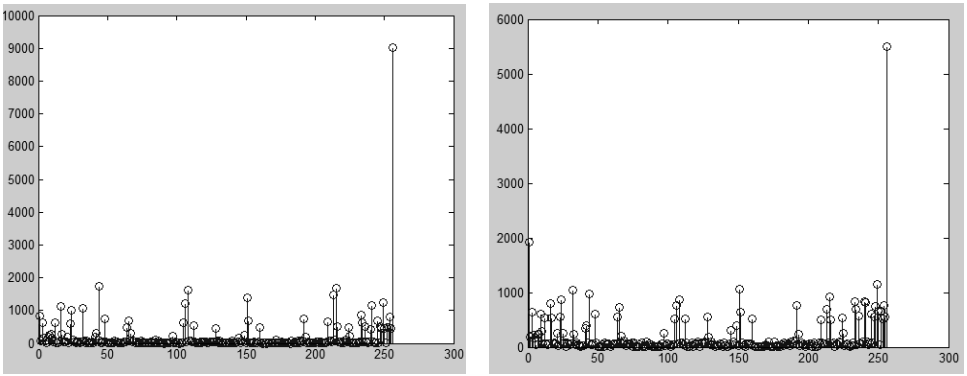
实际试验过程中，传统的 LBP 识别算法在测试中，识别率并不理想，当识别目标超过 5 个之后，识别率就已经下降到了 30%以下，实验分析结果如下。图 3.13 所示是可见光图像和素描图像的 LBP 图像对比。



（a）原图像灰度图　（b）原图像 LBP 图　（c）素描图像　（d）素描图像 LBP 图像

图 3.13　原图像和素描图像的对比

从图 3.13 可以看出，尽管 LBP 图像能够将图像的纹理信息凸显出来，但是由于素描照本身的特性，素描照的 LBP 图像噪声明显多于可见光图像的 LBP 图像，效率并不理想。从图 3.14 可见，在素描照 LBP 直方图中，低灰度值统计数量较大，在 0～50 灰度值范围内尤其明显，而灰度值为 0 的数量很多，这说明了素描图像黑色噪点偏多，从图 3.13（d）中可以清楚地看到许多黑点噪声，而后续滤波器的处理，将会减少图像噪声点，增强纹理信息。



（a）原图像 LBP 直方图　　（b）素描图像 LBP 直方图

图 3.14　LBP 直方图对比

3.6.4　滤波器分析

直接使用 LBP 算法进行识别效果十分糟糕，因为当图像转换为 LBP 图像后，图像中纹理并不清晰，噪点也十分多，LBP 直方图特征也不明显。因此，对图像进行必要的预处理，使用滤波器进行滤波是一种有效的手段。

本章使用了 3 种滤波器进行试验，分别是中值滤波器[24]、高斯滤波器[25]、DOG[26]（Different of Gaussian）滤波器进行尝试。图 3.15 展示了各个滤波器处理后的图像及其对应 LBP 图像。

（a）中值滤波器对比图

（b）高斯滤波器对比图

图 3.15　滤波器处理效果

（c）DOG 滤波器对比图

图 3.15　滤波器处理效果（续）

其中，第 1 列图像是可见光图像经过滤波器处理后的图像，第 2 列图像是对应的 LBP 图像，第 3 列是素描图像经过滤波器处理后的结果，第 4 列是对应的 LBP 图像。可以看出，高斯滤波器和 DOG 滤波器都大大增强了 LBP 图像的效果，而中值滤波器效果则不明显，没有提升图像效果。经过实验验证，在目前数据库中测试，DOG 滤波器的效果是最佳的。

3.7　结合 LBP 和分块特征的识别算法

3.7.1　训练算法

上一节中，已经证明 LBP 算法在识别过程中有一定效果，并且利用 DOG 滤波器对图像加强后，效果进一步加强，然而用整幅图像计算 LBP 直方图，用 Log 概率统计计算两个图像间的相似程度，识别结果依然不能达到理想的结果，因此提出分块的 LBP 算法。在之前的许多人脸识别研究中发现，尽管使用整个人脸的信息量远大于使用局部特征，然而这样同时会带来更多的噪声，反而影响到识别效果，而在使用面部的局部有效特征时，识别率可以有所提升[27]，如眼睛、鼻子、嘴巴等人脸的明显特征。分块 LBP 算法对图像进行分块，之后将每一个小块作为一个可识别特征，利用机器学习思想训练，逐步将特征块进行选取，最后将识别能力最强的特征块取出进行线性组合，最终提升了识别率。具体算法过程如下。

（1）对于 N 个人，每人拥有一张可见光图像，一张素描图像，可见光和素描图像一一对应。

（2）对于所有 $2N$ 个样本，进行 DOG 滤波器处理，得到对应 $2N$ 个 DOG 处理后的图像，之后，再对所有样本进行 LBP 处理，得到对应的 $2N$ 张 LBP 图像。

（3）对图像进行分块处理，块状为方形，边长为 l，每个块平移长度为 step，一旦 l 和 step 确定，则每个图像分出的小块数量可以确定，假设为 y。分块示意图如图 3.16 所示。

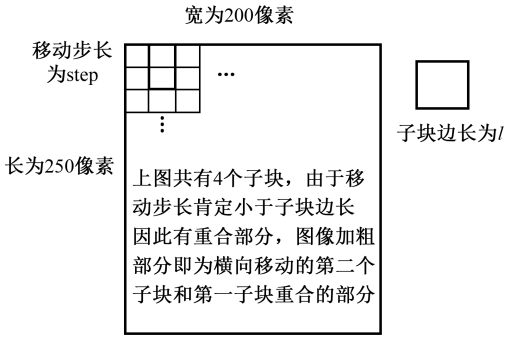


图 3.16　分块示意图

（4）以 photo_i 表示某个照片的图像，sketch_i 表示某个素描照的图像，i=1,2,⋯,N，对于某个人照片图像中的一个子块，以 $\text{photo}_{i,m}$ 表示，素描子块以 $\text{sketch}_{j,n}$ 表示，其中 i,j=1,2,⋯,N，m,n=1,2,⋯,y。

（5）对图像进行分组处理，方便后续的权重判定处理以及子块判定处理。将 1 号素描照和所有可见光图像作为第 1 组，将 2 号素描照和所有可见光图像作为第 2 组，以此类推，可以得到 N 组图像。如图 3.17 所示，图中仅用连线表示了 1 组与 2 组，其他组方法一致，没有全部画出。

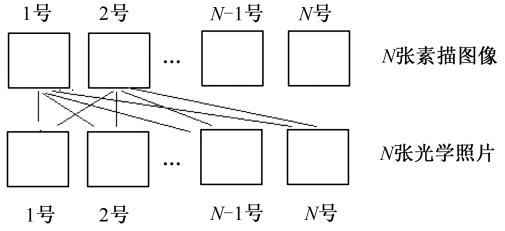


图 3.17　分组示意图

对所有分好组的图像进行权重分配，一般情况下，使用平均权重分类，即每组初始权重为 w_x=1/N，其中 x=1,2,⋯,N。

（6）对所有分组进行循环训练，$w_x = 1/N$；

① 初始化权重为 $w_{1,x} = w_x$，其中 $w_{1,x}$ 为第 1 轮训练的初始化权重。

For $t = 1:T$，其中 T 表示若一共训练 T 轮，t 为当前训练轮次号。

② 权重归一化：

$$q_{t,x} = \frac{w_{t,x}}{\sum_{k=1}^{N} w_{t,k}} \tag{3.7.1}$$

$q_{t,m}$ 为归一化后的权重。这里 x 为组号，t 为轮次号，y 表示取遍所有组号。

③ 对于每一组图像，利用子块计算 LBP 向量进行判别，比如对于第 1 组图像，计算所有图像第 i 号子块的 LBP 向量，接着可以得到所有第 1 组中可见光图像和素描图像第 i 号子块的 LBP 向量距离。若素描照和对应的可见光照片的 LBP 距离是最小的，即第 N 组中，素描照能匹配到第 N 张可见光图像，则第 i 号子块对本组照片判断结果为正确，反之为错误。将所有组的照片都如此处理，可以得到第 i 号子块在 N 组照片中的识别能力。

④ 对于每个子块 i（i=1,2,..,y），用 ε_f^i 表示加权错误率：

$$\varepsilon_f^i = \sum_x q_{t,x} h(d) \tag{3.7.2}$$

其中，$h(d)$ 为这个子块在 x 组照片中判断结果，判断正确为 0，判断错误为 1。

⑤ 选取本轮最佳识别子块

$$\varepsilon_t = \min_d(\varepsilon_f^i) = \min(\varepsilon_f^1, \varepsilon_f^2, \varepsilon_f^3, ..., \varepsilon_f^y) \tag{3.7.3}$$

$\min_d(\varepsilon_f^i)$ 为最本轮加权错误率最低的子块。

⑥ 根据这个子块结果，调整各个组的权重，

$$w_{t+1,x} = w_{t,x} \beta_t^{1-e_x} \tag{3.7.4}$$

其中，$e_x = 0$ 表示本组为正确识别，$e_x = 1$ 表示本组被错误分类，而 $\beta_t = \frac{\varepsilon_t}{1-\varepsilon_t}$。

（7）最终记录子块信息，包括子块顶点坐标和 α_t（$\alpha_t = \log_{10}\frac{1}{\beta_t}$），一般而言，训练轮数决定了最终识别的准确程度，轮数越多，特征越多，识别率相对会提升。关于这一点会在下节更详细地说明。

训练流程图如图 3.18 所示。

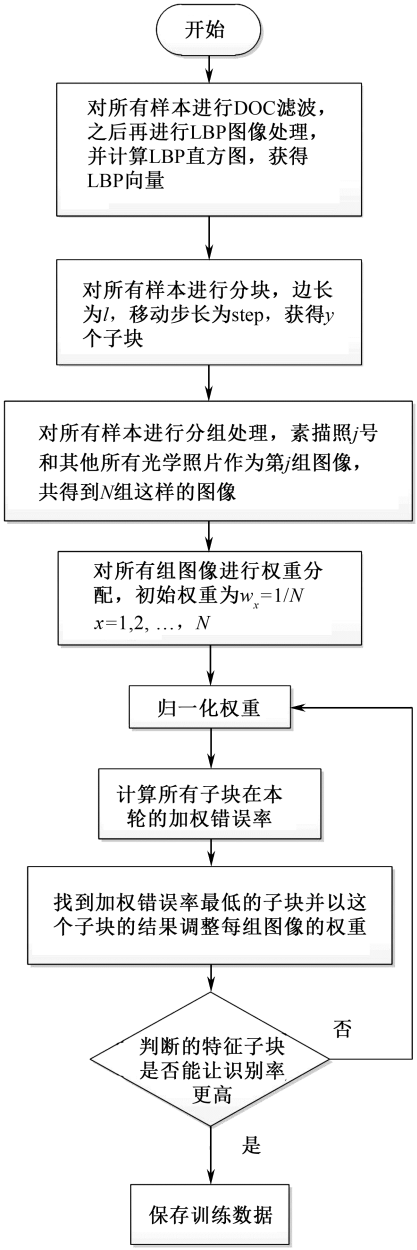


图 3.18　训练算法流程图

3.7.2 识别过程

分块 LBP 算法的识别过程依赖预选好的有效子块进行判决，而由于每个子块在训练中表现出的识别能力也并不相同，因此，需要给每个子块设置判决权重，具体过程如下。

（1）设输入需要识别的素描图像为 X，对识别备选普通照片的数量为 M，用 O_i 表示，$i=1,2,\cdots,M$ 。

（2）如 3.7.1 节中所述，若有 n 个子块进行判决，且对应权重为 $\alpha_t(t=1,2,3,\cdots,n)$。对于输入样本 X，同样可以找到对应的 n 个子块，这样一来，可以计算 X 和任意 O_i 之间的距离。

（3）对于 X 和任意 O_i，以 X_t 和 O_i 表示 2 个图像中的某个对应子块，则计算这两个子块间的 LBP 距离：

$$L_t = F(X_t, O_{i,t}) = \sum_{b=1}^{B} X\_h_{t,b} \log_{10} O\_h_{i,t,b} \tag{3.7.5}$$

其中，X_h、O_h 为两个子块对应的 LBP 向量。

（4）最后 2 个图像间的总距离有式（3.7.6）得出：

$$Lf_i = \sum_{t=1}^{n} (L_t \alpha_t) \tag{3.7.6}$$

其中，α_t 为训练中所得到的子块权重。

（5）计算出所有 O_i 对应的 Lf_i，则当 $I=i, i\in \min(Lf_i)$，则 O_i 为 X 的匹配光学照片。

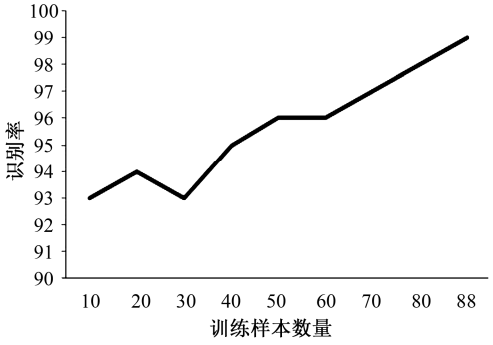
3.8 素描人脸识别实验结果和分析

本实验中使用数据均为 CUHK 中的人脸数据，其中有 188 个人，每个人各有 1 幅照片及 1 幅素描画。将前 88 人的照片及素描画作为训练样本，在后 100 人的照片中对后 100 人的素描画进行识别测试。

3.8.1　训练样本数量分析

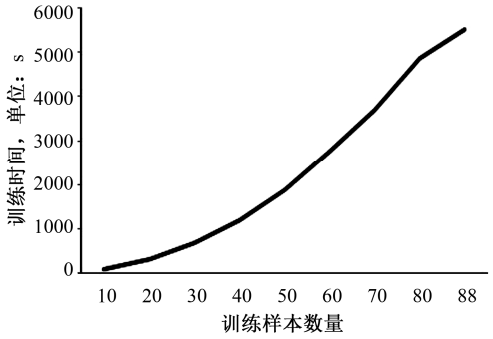
本实验和传统人脸识别有所区别。传统识别中，比如利用 ORL 数据库训练，一个人对应 10 张图像，以其中 8 张作为训练集，测试集则是训练集中这个人剩下的 2 张图像。而本章采用非交叉验证，训练图像和测试图像没有任何交集，比如，训练集中人是 A、B、C 作为训练集，测试集的人则是 D、E、F。因此训练样本的数量的选择将直接影响到结果。理论上，训练样本数量越多，最终效果也会更好，然而样本数量增加也会大大增加训练所消耗的时间成本，因此本章中验证了样本增加对算法效果提升的有效程度。

如图 3.19 所示，训练样本均使用 20×20 子块，步长为 10 的分块方法进行训练，并在识别过程中，对 100 个样本进行识别，并取有效子块数 100。可以看到，在提升训练数量过程中，识别率基本是以上升趋势提升，只有当 20 提升到 30 个样本的时候，出现了异常，可能原因是训练样本数量较小。对应的，随着训练样本的增加，训练时间也大大增加，88 个训练样本达到最佳效果的训练时间为 5517 秒，而使用 10 个样本训练所花时间为 78 秒。从实验结果来看，提升训练样本后识别效果也有显著提升，正确率从 93%提升到了 99%，而训练时间在实际的识别过程中，并不会影响识别速度，因此这个时间成本是值得的，而实验也验证了提升训练样本可以有效提升识别效果。



（a）训练样本数量和识别率的关系

图 3.19　训练样本影响



（b）训练样本数量和训练时间的关系

图 3.19　训练样本影响（续）

3.8.2　特征数量对识别效果的影响

识别过程中，特征数量的多少，对识别结果有着决定性的作用，不仅仅体现在了识别正确率方面，也体现在了识别所需时间上面，使用子块数量越大，识别速度越慢，正确率则越高，反之，识别速度越快，正确率也会大大降低。

实际使用中，我们需要结合实际情况考虑，当系统对实时性要求较高时，必要地减少特征数量，减少识别时间是最为有效的手段，因此需要在识别率和时间中做出综合考虑。本章以 88 个样本训练结果为模板，测试样本集使用 100 个测试目标，测试了特征数量对识别结果的影响。

如图 3.20 所示，随着使用特征的提升，识别效果大大增强。使用一个特征时，识别 100 个样本识别率只有 6%，当提升至 10 个特征时，识别率提升至 76%。而当特征增加到 20 个之后，识别率达到 92%，之后尽管随着特征数量的增大，识别效果有所提升，然而效果并没有开始提升的那么明显。使用 100 个特征时识别率为 99%，比 20 个特征的识别率高出 7 个百分点。从实际使用上来说，识别率提升虽然很多，但是时间成本的增加也不可忽视。因此，使用特征的数量多少需要根据实际需求来决定，如果实时程度要求低，可以使用更多的特征。如果实时程度要求高，则适当减少特征数量，在识别时间和识别率中寻找一个平衡点来达到目标。

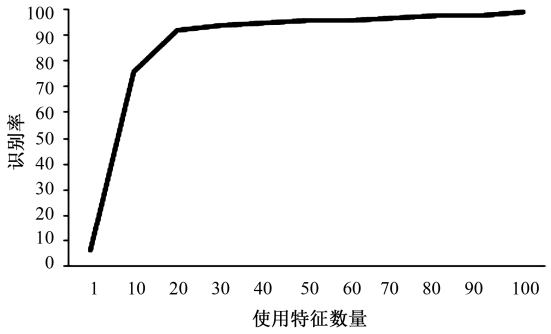


图 3.20　特征数量和识别率的关系

3.8.3　识别级别对识别结果的影响

识别中，识别级别（Rank 值）也是一个很重要的标准。类似于检测中的虚警率，识别级别和识别精度是相互矛盾的一对数据，识别级别越低，识别精度越高，识别准确率会越低，反之识别级别越高，识别精度也越低，识别准确度会越高。如当 Rank=1，则认为和待识别目标距离最小的图像为匹配对象；若 Rank=*N*，则认为和待识别目标距离最小的 *N* 个图像都为匹配对象。其中，只有当待识别目标的真实匹配目标在匹配对象集时，判定为识别正确。显然，当 *N* 越大，识别正确的可能也越高，但是识别出来的干扰目标也越多。

而在实际应用中，提升识别级别可以在更加苛刻的环境中找到需要的识别目标，有效地减少识别范围，本章总结了在各个特征下，不同 Rank 值下的识别效果，同时，将原始的不分块 LBP 算法在不同 Rank 下的识别正确率进行比对。

如图 3.21 所示，随着 Rank 值的增加，各个数量的特征下，识别率是不断上升的，而原始 LBP 算法在不进行分块的情况直接使用，效果十分糟糕。可以发现在特征数量较大时，使用很少的阶数，就可以达到 100%的识别率，如在 100 个特征下，使用 Rank=2，即可以达到 100%的识别率。由以上对比结果可以知道，分块 LBP 特征经过 AdaBoost 算法训练后，比原始 LBP 算法有效许多，同时，增加训练轮数并增加判断的特征数量，也能有效提升识别效果。

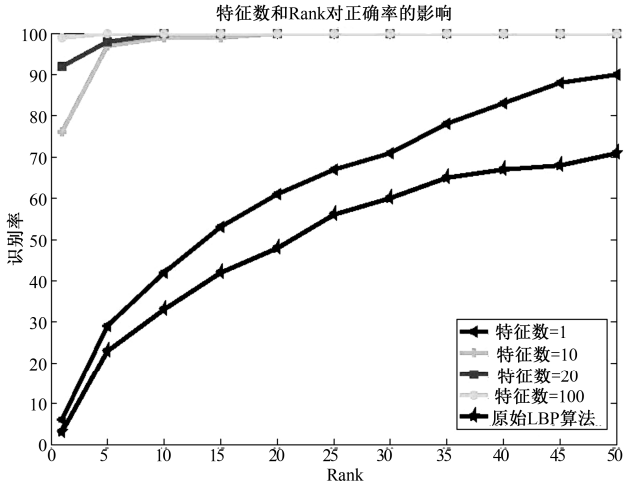


图 3.21　特征数量和 Rank 对正确率的影响

3.8.4　和目前已存在算法进行比较

目前在这项研究中，主要以 Brendan F. Klare 博士的研究成果最为显著[14]，Brendan F. Klare 将常见的各种算法均进行过仿真，并得出结论，之后将这些方法在更实际应用进行过测试。本章所提出的方法因缺乏相应的实验数据，不能全面地和 Brendan F. Klare 的各个算法进行对比，但目前本章中的实验数据仍可以证明本章算法还是行之有效的一个新方法。与 Brendan F. Klare 数据的详细的比对在表 3.3 中列出。

表 3.3　Rank=1 时，算法对比结果

算法名称	特征数	样本数	正确率
DOG+MLBP	59	202	96.3%
CSDN+MLBP	59	202	96.2%
Guassian+SIFT	128	202	97.8%
本章算法	50	100	96%
本章算法	100	100	99%

在表 3.3 中，前 3 个算法都是 Brendan F. Klare 博士的实验结果，其中“+”号之前为滤波器算法，“+”号之后为使用特征算法。图 3.22 中展示了本章算法部分的识别结果，图中一个打叉的图像为本章算法在最好结果中未能识别出的图像。

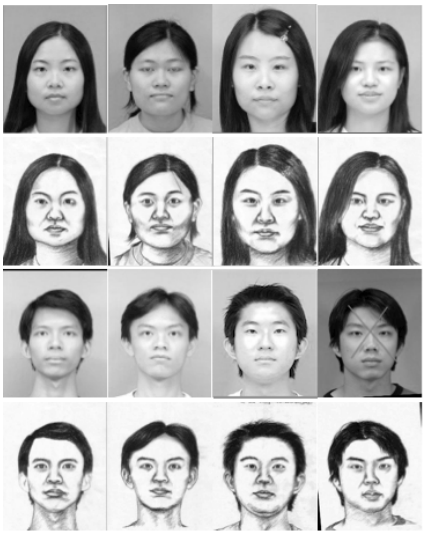


图 3.22　素描人脸识别部分结果

3.8.5　交叉验证实验

为了验证本章算法的有效性，进行了交叉与非交叉验证对比实验。由于 CUHK 数据库中每个人只有 1 张光学图像和 1 张与之对应的素描图像，因此需利用作图工具（本章采用美图秀秀工具），对后 100 人的光学图像进行处理，生成额外的 100 幅近似素描图像，简称美图素描脸。

对于非交叉验证，如 2.1 节所述，用前面 88 人的光学与素描人脸进行训练，用后 100 人的美图素描脸进行测试。对于交叉验证，用后 100 人的光学图像和素描图像进行训练，用后 100 人的美图素描脸进行测试。实验结果如表 3.4 所示。

表 3.4　交叉与非交叉实验比对

特征数量	交叉实验正确率	非交叉实验正确率
特征数=10	96%	95%
特征数=20	98%	96%
特征数=30	100%	100%
特征数=100	100%	100%

由表 3.4 可以看到，无论是交叉验证还是非交叉验证，本章算法均保持了较高的准确率。在交叉验证中，由于训练样本和测试样本来自同一个类别，因此实验结果比非交叉验证略好，但当特征子块数量达到 30 时，两种验证方法的识别

率都达到了 100%。综上所述，本算法在交叉与非交叉验证中，取得良好的实验结果。

3.9 本章小结

本章提出了基于 SIFT 特征的人脸验证算法，现有人脸验证算法，基本是有监督训练过程的机器学习算法，验证的目标需要进行训练才能获得较好的验证结果，并且，大部分算法训练难度高，分类器维度大，验证效率一般。本章提出以 SIFT 特征为基础，划分为数量特征和位置特征进行验证，利用数量特征过滤掉好排除的图像，再利用位置特征得到匹配向量，计算图像相似度，进一步确定匹配的图像，实现了人脸验证。这个方法的优点是没有监督训练过程，可以对各个目标直接进行验证，具有良好的兼容性，适用于更多的验证场合。

同时，还提出了一种新的素描人脸识别，以 LBP 算子描述素描人脸和光学照人脸的相似性，确定 LBP 特征的有效性后，采用 DOG 滤波器进一步提升了图像纹理效果。为了进一步提升识别的准确率，利用机器学习思想，加入分块特征，加以训练，得到能有效分类的分块号和权重。最后的识别中，使用了加权距离概念，利用所有子块的加权距离衡量各个图像之间的相似度，实现了素描人脸识别。本方法和传统的人脸识别方法不同，可以对训练样本以外的图像库进行识别，无交叉验证和交叉验证的实验结果均证明了算法的有效性，是一种新的有效的素描人脸识别的方法。

本章参考文献

[1] Kittler J. Face Authentication Using Client Specific Fisherfaces[EB]. 2001-02-09, Citeseerx. ist.psu.edu/viewdoc/download:jsessionid.

[2] Y P Li. LDA and its application to face identification[D]. Surrey: University of Surrey, 2000.

[3] K Messer, J Kittler, J Luettin, et al. XM2VTSDB: The extended M2VTS database. Audio and Video-based Biometric Person Authentication [C]: Surrey, 1999.

[4] 蔡亮, 达飞鹏.结合形状滤波和几何图像的 3D 人脸识别算法[J]. 中国图象图形学报, 2011, 16(7): 1303～1309.

[5] 袁宁. 人脸验证算法的改进与研究[D]. 无锡：江南大学, 2008.

[6] Shu X, Gao Y,Lu H. Efficient linear discriminant analysis with locality preserving for face recognition [J]. Pattern Recognition, 2012, 45(5): 1892-1898.

[7] Xanthopoulos R Pardalos P M, Trafalls T B. Linear Discriminant Analysis[M]. Robust Data Mining, Springer New York, 2013: 27-33.

[8] Hae Jong Seo, Milanfar. Face Verification Using the LARK Representation[J]. IEEE Transactioins on Information Forensics and Security, 2011, 6(4):1275-1286.

[9] Juan Ramón Troncoso-Pastoriza, Daniel González-Jiménez, Fernando Pérez-González, Fully Private Noninteractive Face Verification. IEEE Transactioins on Information Forensics and Security, 2013,8(7):1101:1114.

[10] 杨晓敏. 图像特征点提取及匹配技术[J]. 光学精密工程, 2009, 17(9): 2279-2280.

[11] 熊英, 马惠敏. 3 维物体 SIFT 特征的提取与应用[J]. 中国图像图形学报, 2010, 15(5): 814-819.

[12] 赵焕利, 王玉德, 张学志, 薛乃玉. 小波变换和特征加权融合的人脸识别[J]. 中国图像图形学报, 2012, 17(12): 1522～1527.

[13] 丘文涛, 赵建, 刘杰. 结合区域分割的 SIFT 图像匹配方法[J]. 液晶与显示, 2012, 27(6): 827-831.

[14] Klare B. Heterogeneous Face Recognition [D]. Lansing: Michigan State University, 2012.

[15] Akgul T. Can an algorithm recognize montage portraits as human faces[J]. IEEE Signal Processing Magazine, 2011, 28(1): 160-158.

[16] Klare B and Jain A. Matching forensic sketches and mug shots to apprehend criminals[J]. IEEE Computer, 2011, 44(5): 94-96.

[17] Klare B and Jain A. On a taxonomy of facial features. Biometrics: Theory Applications and Systems (BTAS), 2010 Fourth IEEE International Conference on Biometrics Compendium[C]. Washington, DC:IEEE, 2010: 1-8.

[18] Ahonen T, Hadid A, and Pietikainen M. Face description with local binary patterns: Application to face recognition [J]. IEEE Trans. Pattern Analysis & Machine Intelligence, 2006, 28(12): 2037 - 2041.

[19] 山世光, 高文, 陈熙霖. 基于面部纹理分布和变形模板的面部特征提取[J]. 软件学报, 2001, 12(4): 570-577.

[20] The CUHK Face Sketch Database is available for download at: http://mmlab.ie.cuhk.edu.hk/facesketch.html

[21] 黄非非. 基于 LBP 的人脸识别研究[D]. 重庆：重庆大学, 2009.

[22] 谢志华, 伍世虔, 方志军. LBP 与鉴别模式结合的热红外人脸识别[J].中国图像图形学报, 2012, 17(6): 707-711.

[23] 高涛, 何明一, 戴玉超, 等. 多级 LBP 直方图序列特征的人脸识别[J]. 中国图像图形学报, 2009, 14(2): 202-207.

[24] 钟灵,章云. 等级阈值的彩色图像适量中值滤波器[J]. 中国图像图形学报, 2011, 16(3): 330-335.

[25] 管涛, 李玲玲. 高斯混合模型、求解算法及视觉应用综述[J]. 中国图象图形学报, 2012, 17(12): 1461-1471.

[26] Tan X and Triggs B. Enhanced local texture feature sets for face recognition under difficult lighting conditions [J]. IEEE Trans. on Image Processing, 2010, 19(6): 1635–1650.

[27] Meyers E and Wolf L. Using biologically inspired features for face processing[J]. Journal of Computer Vision, 2008, 76(1):93-104.

第 4 章

Gabor 小波在人脸识别中的应用研究

本章通过分析 Gabor 小波变换的分类性能，提出了一种基于 Gabor 小波变换和隐马尔可夫模型的人脸识别算法。该算法先对人脸图像进行多分辨率的 Gabor 小波变换，然后在图像上放置一组网格结点，采用主元分析法对每个结点进行去相关、降维，最后形成 Gabor 脸。把 Gabor 脸的每个特征结作为观测向量，对隐马尔可夫模型进行训练，并把优化的模型参数用于人脸识别。之后分析了观测向量维数与识别率的关系，以及状态个数和高斯概率混合成分的个数对识别率的影响，测试了待识别图像经过部分遮挡后，算法的识别性能。对算法复杂度进行了分析，并同其他 4 种相关方法进行了比较。实验结果表明，该方法识别率高，复杂度较低，对部分遮挡的图像具有较大的容忍度。

此外，本章还提出了一种基于 Gabor 小波变换、独立元分析和隐马尔可夫模型的人脸识别方法。独立元分析（Independent Component Analysis，ICA）法可以降低信号统计相关性，它的目标是寻找一个线性变换，把一系列随机变量表达成若干个统计独立的源信号的线性组合。该统计模型用高阶累积量来表示信号的互信息，并通过最小化互信息来确定所需的线性变换。由于 ICA 充分利用了信号的高阶统计量，在降低信号统计相关性的同时，也降低了信号的维数，所以它可以更好地表达信号的本质特征。该方法仍先对人脸图像进行多分辨率的 Gabor 小波变换，但是采用独立元分析法对每个 Gabor 特征网格结点进行去相关、降维，

并形成特征结；然后把每个特征结作为观测向量，对隐马尔可夫模型进行训练，并把优化的模型参数用于人脸识别。实验结果表明，该方法识别率高，工程上易于应用。

4.1 人脸识别典型方法

人脸识别就是用计算机对人脸图像进行特征提取和识别的模式识别技术，它牵涉模式识别、图像处理、计算机视觉、生理学、心理学及认知科学等方面的诸多知识，并与基于其他生物特征的身份鉴别方法，以及计算机人机感知交互领域都有密切联系。本节概述了人脸识别的典型方法，讨论了其中的关键技术和难点及应用和发展前景，并对人脸识别研究中应注意的问题提出了一些看法。

4.1.1 子空间方法

人脸图像的维数通常都是非常高的，而实际上人脸图像在这样高维空间中的分布很不紧凑，因而不利于分类，并且在计算上的复杂度也非常大。为了得到人脸图像的较紧凑分布，Kirby 和 Turk 等首次把主元分析的子空间思想引入到人脸识别中，并获得了较大的成功。随后子空间分析方法就引起了人们的广泛注意，并成为当前人脸识别的主流方法之一。子空间分析的思想就是根据一定的性能目标来寻找一个线性或非线性的空间变换，把原始数据压缩到一个低维子空间，使数据在子空间中的分布更加紧凑，为更好地描述数据提供了手段，另外计算的复杂度也大大降低。目前在人脸识别中得到成功应用的子空间方法有特征脸（Eigenface）方法、线性鉴别分析（Linear Discriminant Analysis，LDA）、独立元分析、核主元分析、奇异值分解等。

4.1.2 基于连接机制的人脸识别方法

基于连接机制的人脸识别方法就是用一个动态连接结构来对人脸进行建模，通过调整权值来达到人脸识别的目的。基于连接机制的方法包括神经网络方法、弹性图匹配方法等。

神经网络在人脸识别中的应用有很长的历史，它有其特殊的适合于人脸识别的优势，它不像其他方法那样要用一套由人确定的规则，同时避免了复杂的特征抽取工作，它能根据有代表性的样本自我学习，具有鲁棒性和自适应性。此外，神经网络以并行方式处理信息，如果能用硬件实现，就能显著提高速度。神经网络方法在人脸识别领域的应用范围很广，除了用于人脸识别外，还适用于性别识别、种族识别等。

弹性图匹配方法[1]是一种基于动态连接结构（Dynamic Link Architecture，DLA）的方法，它将人脸用格状的稀疏图表示，图中的结点用图像位置的 Gabor 小波分解得到的特征向量标记（称为 Jet），图的边用连接结点的距离向量标记。小波特征分析是一种时频分析，若空间一点周围区域的不同频率响应构成该点的特征串，则其高频部分就对应了小范围内的细节，而低频部分则对应了该点周围较大范围内的概貌。因此采用小波变换特征的弹性图匹配方法，既考虑了局部人脸细节，又保留了人脸的空间分布信息，而且它的可变形匹配方式在一定程度上能够容忍人脸从三维到二维投影引起的变形。

4.1.3　隐马尔可夫模型识别方法

在人脸识别方面，Samaria[2]首先将 HMM 用于人脸识别研究，并取得了比较好的效果，他采用的是一维的隐马尔可夫模型。Nefian 等人[3]对 HMM 用于人脸识别方法进行了改进，采用了嵌入式 HMM，提高了识别速度和识别率。

HMM 使用马尔可夫链来模拟信号统计特性的变化，而这种变化又是间接地通过观察序列来描述的，因此 HMM 是一个双重的随机过程。在 HMM 中，结点表示状态，有向边表示状态之间的转移，一个状态可以具有特征空间中的任意特征，对同一特征，不同状态表现出这一特征的概率不同。由于 HMM 是一个统计模型，对于同一个特征序列，可能会对应许多种状态序列，特征序列与状态序列之间的对应关系是非确定的。这种模型对于状态序列来说是隐藏的，故称为隐马尔可夫模型。

4.1.4　基于贝叶斯的人脸识别方法

Moghaddam 和 Pentland 的贝叶斯人脸识别[4,5]把人脸图像之间的变化 Δ 分为两类，即人内（类内）变化 Ω_{I}（同一个人，不同图像之间的变化）和人间（类间）变化 Ω_{E}（不同人的图像之间的变化），从而把人脸识别置于贝叶斯框架之下。

在该方法中，两幅图像 I_1 和 I_2 之间的相似性 $S(I_1,I_2)$ 以概率的形式定义：

$$S(I_1,I_2)=P(\Delta\in\Omega_{\mathrm{I}})=P(\Omega_{\mathrm{I}}\mid\Delta)$$

式中，$P(\Omega_{\mathrm{I}}\mid\Delta)$ 可根据贝叶斯规则，利用条件概率 $P(\Delta\mid\Omega_{\mathrm{I}})$ 和 $P(\Delta\mid\Omega_{\mathrm{E}})$ 计算而得，这两个条件概率则从训练数据中学习得到。在识别时，用最大后验概率（MAP）准则进行判别。这种方法的难点在于，两幅图像之间的差异向量 Δ 的维数 N 很大（$\Delta\in R^N, N=O(10^4)$），因而没有足够的样本来计算条件概率密度的二阶统计值。为了解决这一问题，Moghaddam 和 Pentland 提出了一个对概率密度进行估计的方法：利用特征分解，把向量空间 R^N 划分为两个互补的子空间，一个为主子空间 F，包含 M 个主分量（$M<<N$）；另一个为 F 的正交补空间 $\overline{F}$，它包含其余的残余分量。按照文献[5]中导出的公式，条件概率的估计值可表示为两个独立的高斯边缘分布密度的乘积：

$$\hat{P}(\Delta\mid\Omega)=\left(\frac{\exp(-\frac{1}{2}\sum_{i=1}^{M}\frac{y_i^2}{\lambda_i})}{(2\pi)^{M/2}\prod_{i=1}^{M}\lambda_i^{1/2}}\right)\cdot\left(\frac{\exp(-\frac{\varepsilon^2(\Delta)}{2\rho})}{(2\pi\rho)^{(N-M)/2}}\right)=P_F(\Delta\mid\Omega)\hat{P}_{\overline{F}}(\Delta\mid\Omega)$$

式中，$P_F(\Delta\mid\Omega)$ 是在 F 空间中真正的边缘密度，$\hat{P}_{\overline{F}}(\Delta\mid\Omega)$ 是 $\overline{F}$ 空间中的边缘密度的估计值，y_i 是主成分，λ_i 是对应的特征值，$\varepsilon^2(\Delta)$ 是残余能量，ρ 为 $\overline{F}$ 中特征值的平均值。由条件概率密度的估计值 $\hat{P}(\Delta\mid\Omega)$，以及 $P(\Omega_{\mathrm{E}})$、$P(\Omega_{\mathrm{I}})$ 便可根据贝叶斯规则计算出后验概率 $P(\Omega_{\mathrm{I}}\mid\Delta)$，从而进行识别。

4.1.5 基于流形的人脸识别

2000 年，斯坦福大学的 Tenenbaum[6]和伦敦大学的 Roweis[7]在国际著名的杂志《Science》上分别发表了《基于全局几何结构的非线性降维》和《基于局部线性嵌入的非线性降维》。两篇文章都使用非线性流技术对人脸、手势等大型数据进行降维，部分解决了维数灾难问题。特征提取在人脸识别中的作用至关重要，如何根据人的视觉机制提取有效的特征一直是模式识别领域的研究热点。原始的人脸数据经过传统的主元分析后，其维数确实降低了，但是在低维空间中的欧式距离是否能表达高维空间中各点之间的位置关系呢？答案是否定的，上述两篇文章都表达了如下观点：经过主元分析降维后的数据位于非线性流上，传统的求两点之间的距离是求图中虚线两点之间的“位移”，而正确的求法应该是沿着实线在非线性流上求两点之间的“路程”。文献[8]从视觉认知的角度阐述了相同人的各种姿态在视网膜中所成的像位于非线性流上，可以利用这一点把不同的人用不

同的流形区分，利用曲线拟合技术来比较两个流形之间的相似性。

4.2　隐马尔可夫模型

隐马尔可夫模型（Hidden Markov Model，HMM）是在马尔可夫链的基础上发展起来的。由于实际问题比马尔可夫链所描述的更为复杂，观测到的事件并不是与状态一一对应的，而是通过一组概率分布相联系，这样的模型就称为 HMM。它是一个双重随机过程，其中之一是马尔可夫链，这是基本随机过程，它描述状态的转移。另一个随机过程描述状态和观测值之间的统计对应关系。这样，站在观察者的角度，只能看到观测值，不能直接看到状态，需要通过一个随机过程去感知状态的存在及其特性。

4.2.1　隐马尔可夫模型介绍

隐马尔可夫模型是一种用参数表示的，用于描述随机过程统计特性的概率模型[9]。它由两部分组成：一个是隐含的马尔可夫链，称为隐含层；另一个是实际的观测量，称为观测层。下面以一阶离散马尔可夫过程为例，介绍隐马尔可夫模型的组成，如图 4.1 所示。

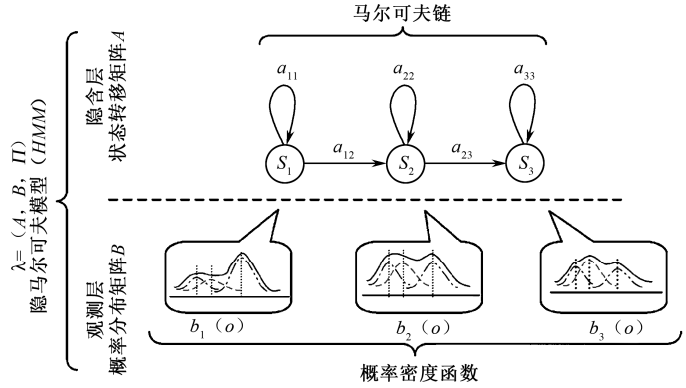


图 4.1　隐马尔可夫模型的组成

（1）N 表示隐含状态数。S 表示隐含状态，即 $S=\{S_1,S_2,\cdots,S_N\}$。模型在时刻 t 的状态用 q_t 表示（$q_t \in S$，$1 \leqslant t \leqslant T$），其中 T 表示观测序列的长度。

（2）M 表示不同观测符号的总数。如果 V 是所有的观测符号集，则有 $V=\{v_1,v_2,\cdots,v_M\}$。

（3）A 表示状态转移矩阵，$A=\{a_{ij}\}$。其中 $a_{ij}=P(q_{t+1}=S_j \mid q_t=S_i)$，$1 \leqslant i,j \leqslant N$，约束条件为：

$$\begin{cases} 0 \leqslant a_{ij} \leqslant 1 \\ \sum_{j=1}^{N} a_{ij}=1 \end{cases} \tag{4.2.1}$$

（4）B 表示观测层的概率分布矩阵，$B=\{b_j(o_t)\}$，$b_j(o_t)=P(o_t=v_k \mid q_t=S_j)$，其中 $1 \leqslant j \leqslant N$，$1 \leqslant k \leqslant M$，$o_t$ 是在时刻 t 的观测符号。

（5）Π 表示初始状态概率分布，$\Pi=\{\pi_i\}$，其中 $\pi_i=P(q_1=S_i)$，$1 \leqslant i \leqslant N$。

HMM 可简记为 $\lambda=(A,B,\Pi)$。对于离散 HMM，观测量是有限符号集中的一个离散符号；对于连续 HMM，观测量只能用一个概率密度函数来刻画。最常用的概率密度函数是混合高斯概率密度函数：

$$b_i(o_t)=P(o_t \mid q_t=S_i,\lambda)=\sum_{l=1}^{M} c_{il} N(o_t \mid \mu_{il},\Sigma_{il})=\sum_{l=1}^{M} c_{il} b_{il}(o_t) \tag{4.2.2}$$

式中，M 表示混合高斯概率密度函数的个数，有别于离散型 HMM 中的观测符号总数“M”；$N(o_t \mid \mu_{il},\Sigma_{il})$ 为均值向量为 μ_{il}，协方差矩阵为 Σ_{il} 的多元高斯分布概率密度函数；c_{il} 为混合系数，满足 $\sum_{l=1}^{M} c_{il}=1$；$b_{il}(o_t)$ 为第 i 个状态第 l 个分量的高斯概率密度函数。图 4.1 是连续 HMM 的结构示意图，图中采用 HMM 的左右模型，在后续的人脸识别中将使用这种模型结构。

4.2.2 隐马尔可夫模型的三个基本问题

将 HMM 应用到实际中，必须解决三个基本问题[9]，分别如下：

① 给定观测序列 $O=o_1,o_2,\cdots,o_T$ 及模型 $\lambda=(A,B,\Pi)$，如何计算由模型 λ 产生 O 的概率评估值 $P(O \mid \lambda)$。解决这个问题的快速算法是前向-后向算法，在人脸识别阶段和人脸训练阶段都需要利用此算法。

② 给定一个观测序列 $O=o_1,o_2,\cdots,o_T$ 和模型 $\lambda=(A,B,\Pi)$，如何选择一个最佳状态序列 $Q^*=q_1^*,q_2^*,\cdots,q_T^*$，以最好地解释观测序列 O。Viterbi 算法可以解

决这个问题，在人脸训练阶段需要使用该算法进行 Viterbi 分割。

③ 给定一个观测序列 $O=o_1,o_2,\cdots,o_T$，如何确定一个 $\lambda=(A,B,\Pi)$，使 $P(O|\lambda)$ 最大。Baum-Welch 算法将解决这个问题，该算法实际上是解决 HMM 训练，即 HMM 参数估计问题。

问题①是一个评价问题，即如何计算给定模型产生观测序列的概率。我们也可以把它看成一个得分问题，即模型与观测序列的匹配程度如何，这一点很重要。例如，如果我们要在多个模型中选择一个最佳模型，问题①的答案就可以帮助我们找到与观测数据匹配的最佳模型。问题②要帮我们揭开模型的隐含部分，找到“正确”的状态序列。必须明确的是所谓“正确”状态可能是不存在的，因此在实际情形中，我们通常使用一个优化准则来解决这个问题。问题③是我们要尽量优化模型参数以最好地解释一个观测序列是如何产生的。用以调整模型参数的观测序列称为训练序列。在大多数的应用中，训练问题都是很重要的。下面将介绍解决上述三个问题的办法。

1．前向-后向算法

若观测序列 $O=o_1,o_2,\cdots,o_T$ 依一定的概率对应于状态序列 $Q=q_1,q_2,\cdots,q_T$，那么其条件概率为：

$$\begin{aligned}P(O|Q,\lambda)&=\prod_{t=1}^{T}P(o_t|q_t,\lambda)\\&=b_{q_1}(o_1)b_{q_2}(o_2)\cdots b_{q_T}(o_T)\end{aligned}\tag{4.2.3}$$

其中假定了在上述条件下各观测序列是相互独立的。模型 λ 所描述的随机过程出现状态序列 Q 的概率为：

$$P(Q|\lambda)=\pi_{q_1}a_{q_1q_2}a_{q_2q_3}\cdots a_{q_{T-1}q_T}\tag{4.2.4}$$

在模型 λ 条件下 O 和 Q 同时发生的概率为：

$$P(O,Q|\lambda)=P(O|Q,\lambda)P(Q|\lambda)\tag{4.2.5}$$

注意到对于 N 状态的模型 λ，出现长为 T 的状态序列应该有 N^T 种可能，要想求给定模型 λ 下出现观测序列 O 的概率，应该对 N^T 种可能求和，即：

$$\begin{aligned}P(O|\lambda)&=\sum_{\forall Q}P(O|Q,\lambda)P(Q|\lambda)\\&=\sum_{q_1,q_2,\cdots,q_T}\pi_{q_1}b_{q_1}(o_1)a_{q_1q_2}b_{q_2}(o_2)\cdots a_{q_{T-1}q_T}b_{q_T}(o_T)\end{aligned}\tag{4.2.6}$$

这样直接计算概率的计算量是十分巨大的，需要 $(2T-1)N^T$ 次乘法，N^T-1 次加法，总计算量为 $2T\cdot N^T$。例如，对于 $N=5$，$T=100$，需要的总计算量为 $2\times100\times5^{100}\approx10^{72}$。所以直接计算是不实用的。所幸的是前向-后向算法

（Forward-Backward）使这个问题的计算量大大降低（仅需 N^2T 次运算）。定义前向变量 $\alpha_t(i)$ 和后向变量 $\beta_t(i)$ 如下：

$$\alpha_t(i)=P(o_1o_2\cdots o_t,q_t=S_i\mid\lambda) \tag{4.2.7}$$

$$\beta_t(i)=P(o_{t+1}o_{t+2}\cdots o_T\mid q_t=S_i,\lambda) \tag{4.2.8}$$

前向算法如下：

（1）初始化

$$\alpha_1(i)=\pi_i b_i(o_1)，\ 1\leqslant i\leqslant N \tag{4.2.9}$$

（2）递推

$$\alpha_{t+1}(j)=\left[\sum_{i=1}^{N}\alpha_t(i)a_{ij}\right]b_j(o_{t+1})，\ 1\leqslant t\leqslant T-1，\ 1\leqslant j\leqslant N \tag{4.2.10}$$

（3）结束

$$P(O\mid\lambda)=\sum_{i=1}^{N}\alpha_T(i) \tag{4.2.11}$$

类似地，后向算法如下：

（1）初始化

$$\beta_T(i)=1，\ 1\leqslant i\leqslant N \tag{4.2.12}$$

（2）递推

$$\beta_t(i)=\sum_{j=1}^{N}a_{ij}b_j(o_{t+1})\beta_{t+1}(j)，\ t=T-1,T-2,\cdots,1，\ 1\leqslant i\leqslant N \tag{4.2.13}$$

（3）结束

$$P(O\mid\lambda)=\sum_{i=1}^{N}\pi_i b_i(o_1)\beta_1(i) \tag{4.2.14}$$

2．Viterbi 算法

给定一个观测序列 $O=o_1,o_2,\cdots,o_T$ 和一个 HMM 的参数 λ，如何选择一个最佳的状态链 $Q^*=q_1^*,q_2^*,\cdots,q_T^*$ 来解释观测序列 O，通常采用的算法是 Viterbi 算法。定义：

$$\delta_t(i)=\max_{q_1,q_2,\cdots,q_{t-1}}P(q_1,q_2,\cdots,q_{t-1},q_t=S_i,o_1,\cdots,o_t\mid\lambda) \tag{4.2.15}$$

表示沿着一条单路径计算前 t 个观测在 t 时刻结束于 S_i 状态的最高概率，那么可以推出关系式：

$$\delta_{t+1}(j)=\max_i\left[\delta_t(i)a_{ij}\right]b_j(o_{t+1}) \tag{4.2.16}$$

我们再用一个二维阵列 $\{\psi_t(j),1\leqslant t\leqslant T,1\leqslant j\leqslant N\}$ 来跟踪记录 $\delta_t(i)$ 在推导过程中的最佳路径。Viterbi 算法的计算步骤如下：

（1）初始化

$$\delta_t(i)=\pi_i b_i(o_1)，\quad 1\leqslant i\leqslant N \tag{4.2.17}$$

$$\psi_1(i)=0 \tag{4.2.18}$$

（2）递推

$$\delta_t(j)=\max_{1\leqslant i\leqslant N}\left[\delta_{t-1}(i)a_{ij}\right]b_j(o_t)，\quad 2\leqslant t\leqslant T，\quad 1\leqslant j\leqslant N \tag{4.2.19}$$

$$\psi_t(j)=\arg\max_{1\leqslant i\leqslant N}\left[\delta_{t-1}(i)a_{ij}\right]，\quad 2\leqslant t\leqslant T，\quad 1\leqslant j\leqslant N \tag{4.2.20}$$

（3）结束

$$P^*=\max_{1\leqslant i\leqslant N}\left[\delta_T(i)\right] \tag{4.2.21}$$

$$q_T^*=\arg\max_{1\leqslant i\leqslant N}\left[\delta_T(i)\right] \tag{4.2.22}$$

（4）路径回溯（最佳状态链的确定）

$$q_t^*=\psi_{t+1}(q_{t+1}^*)，\quad t=T-1,T-2,\cdots,1 \tag{4.2.23}$$

由式（4.2.23）可知，$\psi_t(j)$ 的意义是在 $t-1$ 时刻，哪个状态（从 $\delta_{t-1}(1),\delta_{t-1}(2),\cdots,\delta_{t-1}(N)$ 中选择）对 $\delta_t(j)$ 的生成贡献最大。此外由式（4.2.21）可知，Viterbi 算法不仅可以对观测序列 O 确定一个最佳状态链，而且可以同时求出模型 λ 最佳地产生观测序列 O 的概率 $P(O,Q^*\mid\lambda)$。

Viterbi 算法对观测序列 O 确定一个最佳状态链，这意味着确定了观测序列和各个状态最可能的对应关系。这种对应关系通常称为对观测序列的 Viterbi 分割。

3．Baum-Welch 算法

HMM 参数的优化问题，也就是如何调整模型参数 $\lambda=(A,B,\Pi)$，使 $P(O\mid\lambda)$ 最大。事实上，给定有限长的观测序列作为训练数据，都不可能得到最优的参数估计。通常情况下，我们可以使用 Baum-Welch 算法得到局部最优解。

定义 $\gamma_t(i)$ 为给定模型和观测序列在 t 时刻处于状态 S_i 的概率：

$$\begin{aligned}\gamma_t(i)&=P(q_t=S_i\mid O,\lambda)\\&=\frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N}\alpha_t(i)\beta_t(i)}\end{aligned} \tag{4.2.24}$$

且满足条件：

$$\sum_{i=1}^{N}\gamma_t(i)=1 \tag{4.2.25}$$

定义 $\xi_t(i,j)$ 为在 t 时刻处于状态 S_i，在 $t+1$ 时刻处于状态 S_j 的概率：

$$\xi_t(i,j)=P(q_t=S_i,q_{t+1}=S_j\mid O,\lambda) \tag{4.2.26}$$

用前、后向变量表示为：

$$\xi_t(i,j)=\frac{P(q_t=S_i,q_{t+1}=S_j,O\mid\lambda)}{P(O\mid\lambda)}=\frac{\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)} \tag{4.2.27}$$

显然，$\gamma_t(i)$ 和 $\xi_t(i,j)$ 满足：

$$\gamma_t(i)=\sum_{j=1}^{N}\xi_t(i,j) \tag{4.2.28}$$

如果将 $\gamma_t(i)$ 对某个时间段 t 求和，可以得到在某个时间段内访问状态 S_i 的平均次数，或者等价地，从状态 S_i 转出的期望次数；类似地，将 $\xi_t(i,j)$ 对某个时间段 t 求和，可以得到从状态 S_i 转移到状态 S_j 的期望次数，即：

$$\sum_{t=1}^{T-1}\gamma_t(i)=\text{从状态}\ S_i\ \text{转移到任何其他状态的平均次数} \tag{4.2.29}$$

$$\sum_{t=1}^{T-1}\xi_t(i,j)=\text{从状态}\ S_i\ \text{转移到状态}\ S_j\ \text{的期望次数} \tag{4.2.30}$$

利用以上公式和定义，我们可以给出 HMM 参数的一组估计公式：

$$\pi_i^{\text{new}}=\gamma_1(i)\quad \text{在}\ t=1\ \text{时刻，处于状态}\ S_i\ \text{的概率} \tag{4.2.31}$$

$$a_{ij}^{\text{new}}=\frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)}=\frac{\text{从状态}S_i\text{转移到状态}S_j\text{的平均次数}}{\text{从状态}S_i\text{转移到任何其他状态的平均次数}} \tag{4.2.32}$$

$$b_j^{new}(o_t)=\frac{\sum_{t=1,\text{满足}o_t=v_k}^{T}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)}=\frac{\text{在状态}j\text{观测到符号}v_k\text{的平均次数}}{\text{处于状态}j\text{的平均次数}} \tag{4.2.33}$$

我们定义当前的模型为 $\lambda=(A,B,\Pi)$，并用它来计算式（4.2.31）～式（4.2.33），我们定义重估公式为 $\lambda'=(A',B',\Pi')$。若 $P(O\mid\lambda')>P(O\mid\lambda)$，则模型 λ' 比模型 λ 更优，也就是说，找到一个新模型，它产生观测序列 O 的概率更大。如果在重估计算中重复使用 λ' 代替 λ，我们就能够提高模型产生观测序列 O 的概率，直到满足某种限制。这一重估过程的最终结果称为 HMM 的最大似然估计。

Baum 定义了一个辅助函数：

$$Q(\lambda,\lambda')=\sum_{q\in Q}\log P(O,q\mid\lambda)P(O,q\mid\lambda') \tag{4.2.34}$$

式（4.2.31）～式（4.2.33）可以由式（4.2.34）通过对 λ' 求极大值得到。Baum 等人证明了最大化 $Q(\lambda,\lambda')$ 导致似然概率的增大，即：

$$\max_{\lambda}[Q(\lambda,\lambda')] \Rightarrow P(O\mid\lambda') \geqslant P(O\mid\lambda) \tag{4.2.35}$$

通过若干次迭代后，似然函数收敛到一个极值点。

重估公式可以认为是 EM 算法的实现。其中，E 步是辅助函数 $Q(\lambda,\lambda')$ 的计算，M 步是对 λ' 求最大值。因此在这个特殊问题上，Baum-Welch 算法等价于 EM 算法[10]。

4.2.3　隐马尔可夫模型算法实现中的问题

本节首先讨论用混合高斯概率密度对 HMM 建模的参数估计问题，然后讨论多个观测值训练时的参数估计问题，最后讨论 HMM 计算中的下溢问题。

1．混合高斯概率密度的参数估计

如果观测符号的概率分布 $b_j(o_t)=P(o_t\mid q_t=S_j)$ 是混合高斯分布，这时需要利用 EM 算法求出其均值、协方差矩阵及混合系数。我们定义在 t 时刻观测的向量来自第 i 个状态中的第 l 个组成成分（$q_t=S_i, m_{q_t=i,t}=l$）的概率为

$$\gamma_t(i,l)=\gamma_t(i)\frac{c_{il}b_{il}(o_t)}{b_i(o_t)}=P(q_t=S_i, m_{q_t=S_i,t}=l\mid O,\lambda) \tag{4.2.36}$$

式中，$b_{il}(o_t)=N(o_t\mid\mu_{il},\Sigma_{il})$ 为第 i 个状态第 l 个分量的高斯概率密度函数，且均值向量为 μ_{il}，协方差矩阵为 Σ_{il}；c_{il} 为混合系数。$\gamma_t(i,l)$ 与 $\gamma_t(i)$ 满足：

$$\gamma_t(i)=\sum_{l=1}^{M}\gamma_t(i,l) \tag{4.2.37}$$

利用 EM 算法求出的混合高斯密度参数如下：

$$c_{il}^{\text{new}}=\frac{\sum_{t=1}^{T}\gamma_t(i,l)}{\sum_{t=1}^{T}\gamma_t(i)}=\frac{\sum_{t=1}^{T}P(q_t=S_i, m_{q_t=S_i,t}=l\mid O,\lambda)}{\sum_{t=1}^{T}\sum_{l=1}^{M}P(q_t=S_i, m_{q_t=S_i,t}=l\mid O,\lambda)} \tag{4.2.38}$$

$$\mu_{il}^{\text{new}}=\frac{\sum_{t=1}^{T}\gamma_t(i,l)o_t}{\sum_{t=1}^{T}\gamma_t(i,l)}=\frac{\sum_{t=1}^{T}P(q_t=S_i, m_{q_t=S_i,t}=l\mid O,\lambda)o_t}{\sum_{t=1}^{T}P(q_t=S_i, m_{q_t=S_i,t}=l\mid O,\lambda)} \tag{4.2.39}$$

$$\Sigma_{il}^{\text{new}}=\frac{\sum_{t=1}^{T}\gamma_t(i,l)(o_t-\mu_{il}^{\text{new}})(o_t-\mu_{il}^{\text{new}})^T}{\sum_{t=1}^{T}\gamma_t(i,l)}$$
$$=\frac{\sum_{t=1}^{T}P(q_t=S_i,m_{q_t=S_i,t}=l\mid O,\lambda)(o_t-\mu_{il}^{\text{new}})(o_t-\mu_{il}^{\text{new}})^T}{\sum_{t=1}^{T}P(q_t=S_i,m_{q_t=S_i,t}=l\mid O,\lambda)} \quad (4.2.40)$$

2．多个观测序列进行训练

实际中，经常用多个观测序列来训练出一个 HMM。例如，同一个人有 E 张不同姿态和表情的照片，为了训练出符合这个人的 HMM，就需要把 E 张照片都利用起来对 HMM 进行训练。设第 e 个观测序列为 O^e，$e=1,2,\cdots,E$，其中 $O^e=o_1^e,o_2^e,\cdots,o_T^e$，$O=\{O^1,O^2,\cdots,O^E\}$。这里假定每个观测序列的长度是相等的，因为对于人脸识别来说所用图像的大小是相同的。

假定各个观测序列独立，则

$$P(O\mid\lambda)=\prod_{e=1}^{E}P(O^e\mid\lambda) \quad (4.2.41)$$

利用式（4.2.24）和式（4.2.36）得出：

$$\gamma_t^e(i)=\frac{\alpha_t^e(i)\beta_t^e(i)}{\sum_{i=1}^{N}\alpha_t^e(i)\beta_t^e(i)} \quad (4.2.42)$$

$$\gamma_t^e(i,l)=\gamma_t^e(i)\frac{c_{il}b_{il}(o_t^e)}{b_i(o_t^e)} \quad (4.2.43)$$

由式（4.2.27）得：

$$\xi_t^e(i,j)=\frac{\alpha_t^e(i)a_{ij}b_j(o_{t+1}^e)\beta_{t+1}^e(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t^e(i)a_{ij}b_j(o_{t+1}^e)\beta_{t+1}^e(j)} \quad (4.2.44)$$

利用 EM 算法（即 Baum-Welch 算法）得出多个观测序列的参数估计公式如下：

$$\pi_i^{\text{new}}=\frac{\sum_{e=1}^{E}\gamma_1^e(i)}{E} \quad (4.2.45)$$

$$a_{ij}^{\text{new}}=\frac{\sum_{e=1}^{E}\sum_{t=1}^{T_e-1}\xi_t^e(i,j)}{\sum_{e=1}^{E}\sum_{t=1}^{T_e-1}\gamma_t^e(i)}=\frac{\sum_{e=1}^{E}\sum_{t=1}^{T_e-1}\xi_t^e(i,j)}{\sum_{j=1}^{N}\sum_{e=1}^{E}\sum_{t=1}^{T_e-1}\xi_t^e(i,j)} \tag{4.2.46}$$

$$c_{il}^{\text{new}}=\frac{\sum_{e=1}^{E}\sum_{t=1}^{T_e}\gamma_t^e(i,l)}{\sum_{e=1}^{E}\sum_{t=1}^{T_e}\gamma_t^e(i)}=\frac{\sum_{e=1}^{E}\sum_{t=1}^{T_e}\gamma_t^e(i,l)}{\sum_{l=1}^{M}\sum_{e=1}^{E}\sum_{t=1}^{T_e}\gamma_t^e(i,l)} \tag{4.2.47}$$

$$\mu_{il}^{\text{new}}=\frac{\sum_{e=1}^{E}\sum_{t=1}^{T_e}\gamma_t^e(i,l)o_t^e}{\sum_{e=1}^{E}\sum_{t=1}^{T_e}\gamma_t^e(i,l)} \tag{4.2.48}$$

$$\begin{aligned}\Sigma_{il}^{\text{new}}&=\frac{\sum_{e=1}^{E}\sum_{t=1}^{T_e}\gamma_t^e(i,l)(o_t^e-\mu_{il}^{\text{new}})(o_t^e-\mu_{il}^{\text{new}})^T}{\sum_{e=1}^{E}\sum_{t=1}^{T_e}\gamma_t^e(i,l)}\\&=\frac{\sum_{e=1}^{E}\sum_{t=1}^{T_e}\gamma_t^e(i,l)o_t^e o_t^{eT}}{\sum_{e=1}^{E}\sum_{t=1}^{T_e}\gamma_t^e(i,l)}-\mu_{il}^{\text{new}}\mu_{il}^{\text{new}\,T}\end{aligned} \tag{4.2.49}$$

3．下溢问题

（1）伸缩因子

在前向-后向算法和 Baum-Welch 算法中，都需要使用前、后向变量$\alpha_t(i)$和$\beta_t(i)$。为了计算$\alpha_t(i)$，可把式（4.2.10）代入式（4.2.7），得到$\alpha_t(i)$由许多项的和构成，每项可表示为[11]：

$$\prod_{s=1}^{t-1}a_{q_s,q_{s+1}}\prod_{s=1}^{t}b_{q_s}(o_s) \tag{4.2.50}$$

由于这些乘积的每项都小于 1，会使结果迅速趋向于零，最终$\alpha_t(i)$的动态范围将超出计算机所能表示的精度范围，从而产生溢出，使计算无法继续进行。同理，计算$\beta_t(i)$也会产生此类问题。为了解决这种下溢问题，需要对$\alpha_t(i)$和$\beta_t(i)$增加伸缩因子，对计算进行补偿，但不改变最后的计算结果。

用$\alpha_t(i)$表示未伸缩时的值，$\hat{\alpha}_t(i)$表示伸缩后的值，$\hat{\hat{\alpha}}_t(i)$是一个中间量，表示用$\hat{\alpha}_{t-1}(i)$计算的结果，首先进行初始化：

$$\hat{\hat{\alpha}}_1(i)=\alpha_1(i) \tag{4.2.51}$$

$$c_1=\frac{1}{\sum_{i=1}^{N}\alpha_1(i)} \tag{4.2.52}$$

$$\hat{\alpha}_1(i)=c_1\alpha_1(i) \tag{4.2.53}$$

并设

$$\hat{\hat{\alpha}}_t(i)=\sum_{j=1}^{N}\hat{\alpha}_{t-1}(j)a_{ji}b_i(o_t)\text{，}\quad 2\leqslant t\leqslant T \tag{4.2.54}$$

给定

$$\hat{\alpha}_t(i)=c_t\hat{\hat{\alpha}}_t(i)$$

则伸缩系数c_t为

$$c_t=\frac{1}{\sum_{i=1}^{N}\hat{\hat{\alpha}}_t(i)} \tag{4.2.55}$$

由式（4.2.54）和式（4.2.55）得：

$$\hat{\alpha}_t(i)=\frac{\sum_{j=1}^{N}\hat{\alpha}_{t-1}(j)a_{ji}b_i(o_t)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\hat{\alpha}_{t-1}(j)a_{ji}b_i(o_t)} \tag{4.2.56}$$

通过对式（4.2.54）和式（4.2.55）递推可以得到：

$$\hat{\alpha}_{t-1}(j)=\left(\prod_{r=1}^{t-1}c_r\right)\alpha_{t-1}(j) \tag{4.2.57}$$

把式（4.2.57）代入式（4.2.56）得到：

$$\hat{\alpha}_t(i)=\frac{\sum_{j=1}^{N}\hat{\alpha}_{t-1}(j)a_{ji}b_i(o_t)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\hat{\alpha}_{t-1}(j)a_{ji}b_i(o_t)}=\frac{\sum_{j=1}^{N}\alpha_{t-1}(j)\left(\prod_{r=1}^{t-1}c_r\right)a_{ji}b_i(o_t)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_{t-1}(j)\left(\prod_{r=1}^{t-1}c_r\right)a_{ji}b_i(o_t)}=\frac{\alpha_t(i)}{\sum_{i=1}^{N}\alpha_t(i)} \tag{4.2.58}$$

由式（4.2.58）可看出，$\hat{\alpha}_t(i)$对$\alpha_t(i)$进行了有效的伸缩。

此外由式（4.2.57）得到：

$$\hat{\alpha}_T(i)=\left(\prod_{t=1}^{T}c_t\right)\alpha_T(i) \tag{4.2.59}$$

所以

$$\sum_{i=1}^{N}\hat{\alpha}_T(i)=\prod_{t=1}^{T}c_t\sum_{i=1}^{N}\alpha_T(i)=1 \tag{4.2.60}$$

由式（4.2.11）和式（4.2.60）可推出：

$$\prod_{t=1}^{T} c_t P(O \mid \lambda) = 1$$

所以

$$\log P(O \mid \lambda) = -\sum_{t=1}^{T} \log c_t \tag{4.2.61}$$

可见在实际计算时，通过计算伸缩因子，可以得到对数评估值 $\log P(O \mid \lambda)$ 。

对于 $\beta_t(i)$ 的伸缩，也可采用如下方法。

初始化：

$$\hat{\beta}_T(i) = \hat{\hat{\beta}}_T(i) = \beta_T(i) = 1, \quad s_T = 1 \tag{4.2.62}$$

并设

$$\hat{\hat{\beta}}_t(i) = \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \hat{\beta}_{t+1}(j), \quad t = T-1, T-2, \cdots, 1 \tag{4.2.63}$$

给定

$$\hat{\beta}_t(i) = s_t \hat{\hat{\beta}}_t(i), \quad t = T-1, T-2, \cdots, 1 \tag{4.2.64}$$

则伸缩系数为

$$s_t = \frac{1}{\sum_{i=1}^{N} \hat{\hat{\beta}}_t(i)} \tag{4.2.65}$$

由式（4.2.62）～式（4.2.64）得：

$$\hat{\beta}_t(i) = \left(\prod_{k=t}^{T} s_k \right) \beta_t(i) \tag{4.2.66}$$

对于 HMM 参数估计问题，要想使伸缩后的参数估计保持不变，根据式（4.2.45）～式（4.2.49），必须使伸缩后的 $\gamma_t(i)$ 、$\xi_t(i,j)$ 和 $\gamma_t(i,l)$ 保持不变，假设伸缩后它们变为 $\hat{\gamma}_t(i)$ 、$\hat{\xi}_t(i,j)$ 和 $\hat{\gamma}_t(i,l)$ ，则

$$\begin{aligned} \hat{\gamma}_t(i) &= \frac{\hat{\alpha}_t(i)\hat{\beta}_t(i)}{\sum_{i=1}^{N} \hat{\alpha}_t(i)\hat{\beta}_t(i)} = \frac{\left(\prod_{k=1}^{t} c_k \right) \alpha_t(i) \left(\prod_{k=t}^{T} s_k \right) \beta_t(i)}{\sum_{i=1}^{N} \left(\prod_{k=1}^{t} c_k \right) \alpha_t(i) \left(\prod_{k=t}^{T} s_k \right) \beta_t(i)} \\ &= \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)}, \quad t = T-1, T-2, \cdots, 1 \end{aligned} \tag{4.2.67}$$

对于 $t = T$ ，直接把 $\hat{\beta}_T(i) = \beta_T(i) = 1$ 代入即可，所以最后得到：

$$\hat{\gamma}_t(i)=\gamma_t(i)，\quad t=T,T-1,\cdots,1 \tag{4.2.68}$$

而对于 $\hat{\xi}_t(i,j)$ 有：

$$\hat{\xi}_t(i,j)=\frac{\hat{\alpha}_t(i)a_{ij}b_j(o_{t+1})\hat{\beta}_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\hat{\alpha}_t(i)a_{ij}b_j(o_{t+1})\hat{\beta}_{t+1}(j)}=\frac{\left(\prod_{k=1}^{t}c_k\right)\alpha_t(i)a_{ij}b_j(o_{t+1})\left(\prod_{k=t}^{T}s_k\right)\beta_t(i)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\prod_{k=1}^{t}c_k\right)\alpha_t(i)a_{ij}b_j(o_{t+1})\left(\prod_{k=t}^{T}s_k\right)\beta_t(i)}$$

$$=\frac{\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)}，\quad t=T-1,T-2,\cdots,1 \tag{4.2.69}$$

对于 $t=T$，直接把 $\hat{\beta}_T(i)=\beta_T(i)=1$ 代入即可，最后得到：

$$\hat{\xi}_t(i,j)=\xi_t(i,j)，\quad t=T,T-1,\cdots,1 \tag{4.2.70}$$

同理

$$\hat{\gamma}_t(i,l)=\hat{\gamma}_t(i)\frac{c_{il}b_{il}(o_t)}{b_i(o_t)}$$

$$=\gamma_t(i)\frac{c_{il}b_{il}(o_t)}{b_i(o_t)}=\gamma_t(i,l) \tag{4.2.71}$$

综上所述，由于伸缩后 $\hat{\gamma}_t(i)=\gamma_t(i)$，$\hat{\xi}_t(i,j)=\xi_t(i,j)$，$\hat{\gamma}_t(i,l)=\gamma_t(i,l)$，所以伸缩不影响参数估计的求解。

（2）对数求解

在 Viterbi 算法中，也存在类似的溢出问题。解决这个问题的一种方法是仍采用伸缩因子，但是比较麻烦。我们观察到在 Viterbi 算法中都是乘法，没有加法，因此可以利用对数把乘法变为加法，从而解决溢出问题。对数求解的具体步骤如下[11]。

① 预处理。

$$\hat{\pi}_i=\log(\pi_i)，\quad 1\leqslant i\leqslant N$$

$$\hat{b}_i(o_t)=\log(b_i(o_t))，\quad 1\leqslant i\leqslant N，\quad 1\leqslant t\leqslant T$$

$$\hat{a}_{ij}=\log(a_{ij})，\quad 1\leqslant i,j\leqslant N$$

② 初始化。

$$\hat{\delta}_t(i)=\log(\delta_t(i))=\hat{\pi}_i+\hat{b}_i(o_1)，\quad 1\leqslant i\leqslant N$$

$$\psi_1(i)=0，\quad 1\leqslant i\leqslant N$$

③ 递推。

$$\hat{\delta}_t(j)=\log(\delta_t(j))=\max_{1\leqslant i\leqslant N}\left[\hat{\delta}_{t-1}(i)+\hat{a}_{ij}\right]+\hat{b}_j(o_t)，\quad 2\leqslant t\leqslant T，\quad 1\leqslant j\leqslant N$$

$$\psi_t(j)=\arg\max_{1\leqslant i\leqslant N}\left[\hat{\delta}_{t-1}(i)+\hat{a}_{ij}\right],\quad 2\leqslant t\leqslant T,\quad 1\leqslant j\leqslant N$$

④ 结束。

$$P^*=\max_{1\leqslant i\leqslant N}\left[\hat{\delta}_T(i)\right] \tag{4.2.72}$$

$$q_T^*=\arg\max_{1\leqslant i\leqslant N}\left[\hat{\delta}_T(i)\right]$$

⑤ 路径回溯（最佳状态链的确定）。

$$q_t^*=\psi_{t+1}(q_{t+1}^*),\quad t=T-1,T-2,\cdots,1$$

由式（4.2.72）可知，采用对数 Viterbi 算法，最后可求出模型的对数评估值：

$$P^*=\log P(O,Q^*\mid\lambda)$$

4.3　基于 Gabor 脸和 HMM 的人脸识别方法

本节提出了一种基于 Gabor 小波变换和隐马尔可夫模型的人脸识别算法。该算法利用 Gabor 小波在不同方向和尺度的伸缩特性，在图像上放置一组网格结点，采用主元分析法对每个结点进行去相关、降维，最后形成 Gabor 脸。把 Gabor 脸的每个特征结作为观测向量，对隐马尔可夫模型进行训练，并把优化的模型参数用于人脸识别。实验结果表明，该方法识别率高，复杂度较低，对部分遮挡的图像具有较大的容忍度。

4.3.1　研究背景

特征提取在人脸识别中的作用至关重要，如何根据人的视觉机制提取有效的特征一直是模式识别领域的研究热点。早期的研究中有人用 Gabor 小波对大脑皮层的视觉感知细胞的性态进行建模[12~14]，即可以把每个视觉细胞看成一个具有一定方向和尺度的 Gabor 滤波器。当外界刺激（如图像信号）输入视觉细胞时，视觉细胞的输出响应就是图像与 Gabor 滤波器的卷积，而这个输出信号经大脑的进一步处理后形成最后的认知映像。由于这种模型能较好地解释人的视觉对图像尺度的伸缩和方向变化的容忍度，因此它被广泛地应用到人脸识别中。1992 年以前，Gabor 小波主要应用在纹理描述、图像分割等领域。1993 年以后，相关研究人员开始把它应用到图像识别领域。目前 Gabor 小波应用到图像识别领域主要

有以下两个研究方向。

1．基于结点的动态连接结构

在人脸图像上放置一组网格结点，每个结点的特征用该结点处的多尺度Gabor 幅度特征描述，从而在每个结点处形成一个特征矢量。Lades 首先通过动态连接机制把 Gabor 小波应用到人脸识别中[1]。在这种方法中，各结点之间的连接关系用几何距离表示，从而形成了基于二维拓扑图的人脸描述，根据两个图像中各结点和连接之间的相似性可以进行人脸识别。Wiskott 对动态连接结构进行了改进，提出了弹性图匹配法[15]，该方法将人脸特征上的一些点作为基准点，强调了人脸关键特征点的重要性，并且在识别时有效地利用了 Gabor 小波的相位信息。该技术在 FERET 测试中[16]若干指标名列前茅，其缺点是时间复杂度高，实现复杂。Duc[17]把简化的弹性图应用到人脸鉴别上，取得了很好的效果。Kruger[18]对弹性图进行了推广，把 Gabor 小波和径向基函数神经网络结合起来，提出了小波神经网络，并把它应用到人脸的姿态估计。总之，上述方法都是在图像上放置一些结点，并利用 Gabor 小波的特性在每个结点处形成一个特征矢量。

2．利用 Gabor 小波变换后的整体特征

Donato[19]首先把 Gabor 小波的整体特征应用到人脸表情的识别。这种方法首先把每个人脸图像与 40 个不同方向和不同尺度的 Gabor 滤波器进行卷积，然后把每个滤波图像按照行或列的顺序串接成一个列向量，最后把 40 个滤波图像串接成一个更大的列向量，这个列向量有效地表示了这张人脸的特征。由于这个列向量的维数很高，可以先对它进行抽样，接着归一化，然后利用 PCA 等方法进行降维。与此类似，还可以把核 PCA[20]、增强的 Fisher 线性判决准则[21]同 Gabor 的整体特征结合起来，同样产生了很好的识别效果。

本节在第一阶段特征提取时主要采用上述第一种方案，但也借鉴了第二种方案中降维的思想，采用 PCA 方法对每个结点形成的特征矢量进行去相关、降维。众所周知，一种人脸识别方法的效果如何，取决于它在多大程度上利用了图像的原始信息，于是可以利用一组数值特征来描述人脸的各个器官，并且利用这组数值特征对人脸进行识别。但是简单地利用一组数值特征不能满意地解决人脸识别问题。视觉识别人脸的机制是十分微妙的，人们对此的认识还非常肤浅。事实上，人脸应当作为一个整体来描述，不仅仅包括各个器官的数值特征，还应当包括各个器官的不同表象和相互关联。弹性图匹配为人脸各个器官的连接提供了一个良

好的机制，但是这种连接机制的复杂度较高。而用于语音识别的隐马尔可夫模型[9]则为我们提供了解决这一问题的另外机制，按照这种模型，观测到的特征被看成是另外不可观测 “状态”产生的一系列实现。因而可以将不同的人用不同的 HMM 参数来表征，而同一个人由于姿态和表情变化产生的多个观测序列可以通过同一个 HMM 来表征，这在理论上有了很大的进步。

利用 HMM 对人脸进行描述和识别，就不是孤立地利用各个器官的数值特征，而是把这些特征和一个状态转移模型联系起来。Samaria[22]最早提出了关于人脸的 HMM，他用一个矩形窗从上到下对人脸图像进行采样，将窗内的像素点排列成向量，用灰度值作为观测向量。Nefian[3]发展了 Samaria 的方法，提出了基于 2D-DCT 特征提取的方法，他用 39 个 2D-DCT 系数作为观测向量，代替 Samaria 的灰度值序列，这在一定程度上解决了 Samaria 的大存储量的缺陷。Nefian 还提出了嵌入式隐马尔可夫模型（Embedded Hidden Markov Model，EHMM），这是一种伪二维模型[3]，有效地提高了识别率，但是计算量增大。Othman[23]则在一定假设的前提下，提出了低复杂度的 2D-HMM 模型，更好地描述了人脸各个器官之间的关联，具有很高的识别率。与完整的二维 HMM 模型相比，Othman 所构建的模型确实降低了计算复杂度，但是同 1D-HMM 相比，其计算复杂度提高很多。

自从 Nefian[3]采用在矩形抽样窗口进行 2D-DCT 来提取 HMM 所需要的观测向量以来，大部分学者在利用 HMM 进行人脸识别时，都采用在矩形窗口进行某种变换来提取特征，常见的变换有 2D-DCT、KLT、小波变换、奇异值分解等。当 1D-HMM 识别率不高时，就采用更复杂的 EHMM 或 2D-HMM，研究的热点主要是如何简化这些复杂的模型，而忽略了对有效特征抽取的研究。本节借鉴了弹性图匹配中特征抽取的思想，在人脸图像上放置一组网格结点，每个结点的特征用该结点处的多尺度 Gabor 幅度特征描述，从而在每个结点处形成一个特征矢量，并把这个特征矢量作为 HMM 的观测向量，然后用 1D-HMM 来描述这些结点之间的关联。实验结果表明，与 EHMM 相比，本节所述的方法与其识别率相当，但由于仅采用了 1D-HMM，因此计算复杂度低。

4.3.2　Gabor 小波概述

由于 Gabor 小波与大脑皮层的视觉感知细胞相关，因此 Gabor 小波被应用到图像分析和人脸识别中。本节采用多分辨率的 Gabor 小波来提取特征，这是因为 Gabor 小波具有如下优点：

① 就最小化空间域和频率域的联合二维不确定性来讲，Gabor 小波是最优的。Gabor 小波表示了这样一种直观概念，当纹理较细致时，空间域采样范围应较小，频率域采样范围应较大；当纹理比较粗糙时，空间域采样范围应较大，频率域采样范围应较小。

② Gabor 小波的方向和尺度可调。

Gabor 小波的核函数定义如下[1]：

$$\psi_{\nu,\mu}(\vec{z})=\frac{\left\|\vec{k}_{\nu,\mu}\right\|^2}{\sigma^2}\mathrm{e}^{-\frac{\left\|\vec{k}_{\nu,\mu}\right\|^2\|\vec{z}\|^2}{2\sigma^2}}\left(\mathrm{e}^{i\vec{k}_{\nu,\mu}\vec{z}}-\mathrm{e}^{-\sigma^2/2}\right) \tag{4.3.1}$$

$$\vec{k}_{\nu,\mu}=\left(k_\nu\cos\varphi_\mu,k_\nu\sin\varphi_\mu\right) \tag{4.3.2}$$

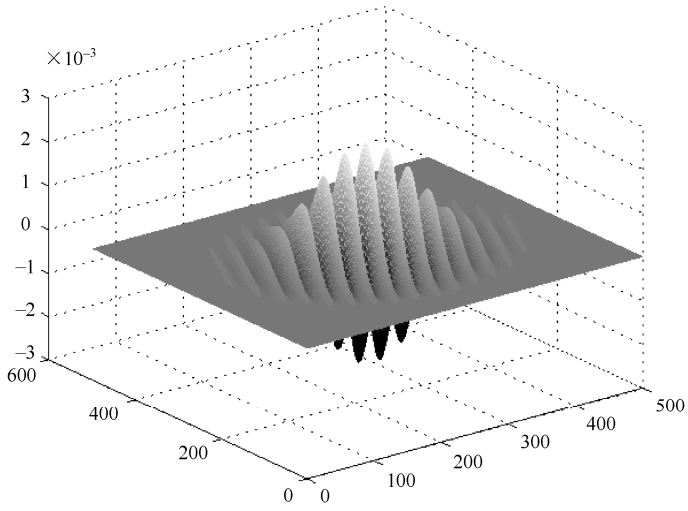
式中，φ_μ 和 k_ν 分别定义了波向量 $\vec{k}_{\nu,\mu}$ 的方向和尺度，$\vec{z}=(x,y)$，$\|g\|$ 定义了向量范数。在式（4.3.2）中，$k_\nu=k_{\max}/f^\nu$，$\varphi_\mu=\pi\mu/8$。f 为频率域中的采样步长，通常取 $f=\sqrt{2}$。$k_{\max}$ 对应最大的采样频率，取 $k_{\max}=\pi/2$。参数 σ 决定了高斯窗的宽度与波向量长度的比率，在本节中取 $\sigma=2\pi$。图 4.2 显示了 Gabor 核函数的实部和虚部波形，结合式（4.3.1）可以看出 Gabor 核函数是一个被复正弦函数调制的高斯窗函数。

各种不同尺度和方向的 Gabor 核函数是自相似的，因为它们都可以通过对母函数中的波向量 $\vec{k}_{\nu,\mu}$ 进行幅度和方向的变化得到。式（4.3.1）的第一部分决定了核函数的振荡波形；第二部分用来补偿直流分量的影响，当参数 σ 很大时，$\mathrm{e}^{-\sigma^2/2}\approx 0$，此时就可以忽略直流分量的影响。在人脸识别中，通常取 5 种不同的尺度和 8 种不同的方向，即 $\nu\in\{0,\cdots,4\}$，$\mu\in\{0,\cdots,7\}$。图 4.3 显示了在 5 种尺度和 8 种方向下，取上述参数时，Gabor 小波的实部和幅度示意图。由图中可看出 Gabor 小波在空间的方向选择性，以及局部幅度的可伸缩性。例如，从图 4.3（a）中的某行可看出每个 Gabor 小波的方向是不同的；从图 4.3（b）可看出空间高斯采样窗口由小到大，相应的采样频率由大到小。此外，图 4.3 也显示了 Gabor 小波在空间域的自相似性。

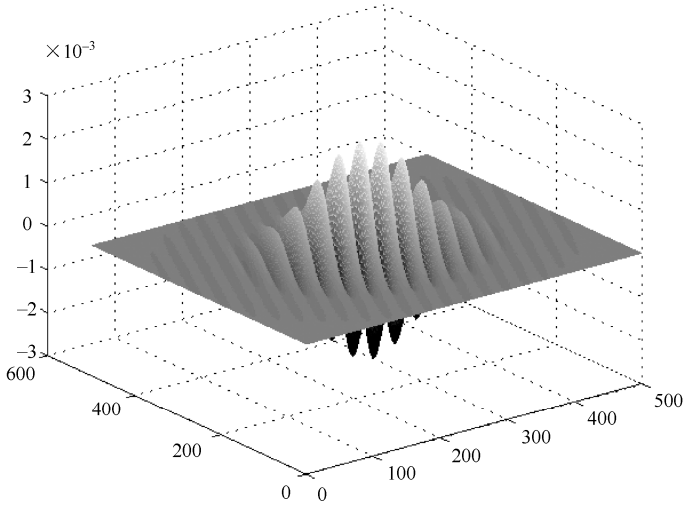
若对式（4.3.1）做傅里叶变换，可以求出 Gabor 函数的频域表达式：

$$F_\psi(\vec{w})=2\pi\mathrm{e}^{-\frac{\sigma^2}{2}}\mathrm{e}^{-\frac{\sigma^2\|\vec{w}\|^2}{2\left\|\vec{k}_{\nu,\mu}\right\|^2}}\left(\mathrm{e}^{\frac{\sigma^2\vec{k}_{\nu,\mu}\vec{w}}{\left\|\vec{k}_{\nu,\mu}\right\|^2}}-1\right) \tag{4.3.3}$$

$$\vec{w}=(w_x,w_y) \tag{4.3.4}$$

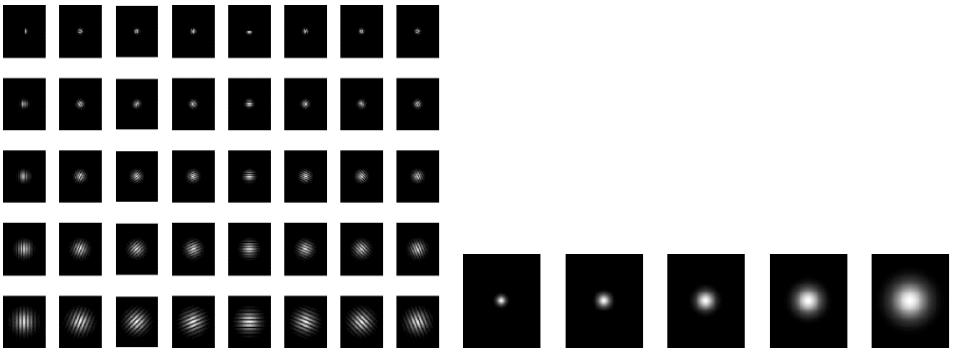


（a）Gabor核函数实部波形



（b）Gabor核函数虚部波形

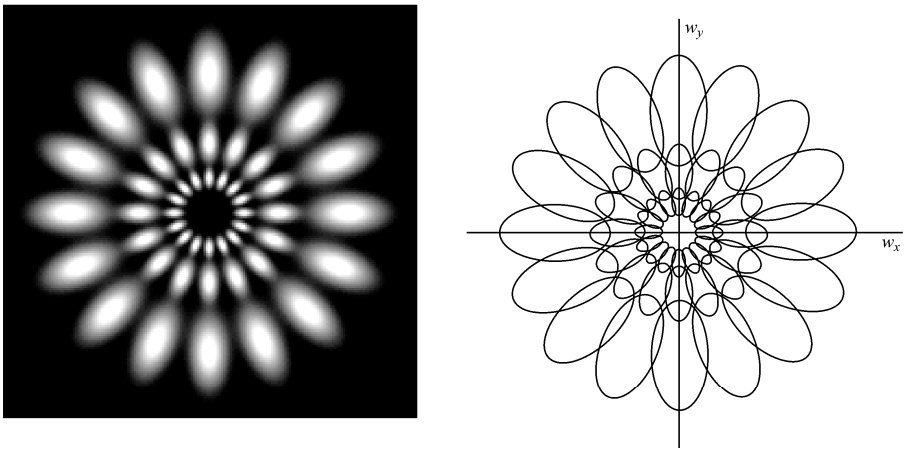
图 4.2　$k_\nu = 0.7854$，$\varphi_\mu = 45^\circ$ 时 Gabor 核函数波形



（a）Gabor 小波的实部　　（b）Gabor 小波的幅度

图 4.3　Gabor 小波在空间域的表现

图 4.4 显示了根据式（4.3.3）画出的 Gabor 小波在频率域的表现。其中，图 4.4（a）表示了不同尺度和方向的 Gabor 小波在频率域的覆盖，可见 Gabor 小波能够近似遍布全频域；而图 4.4（b）显示了每个小波的 3dB 带宽。图 4.4 也显示了 Gabor 小波在频率域中的自相似性。



（a）Gabor小波在频率域的覆盖　　（b）每个椭圆显示了Gabor小波的3dB带宽

图 4.4　Gabor 小波在频率域的表现

4.3.3　利用 Gabor 小波进行特征提取

用 $I(\vec{z}) = I(x, y)$ 表示图像的灰度分布，则图像 I 与 Gabor 小波 $\psi_{\mu,\nu}$ 的卷积为：

$$O_{\nu,\mu}(\vec{z}) = I(\vec{z}) \oplus \psi_{\nu,\mu}(\vec{z}) \tag{4.3.5}$$

式中，符号⊕表示二维卷积。一幅图像经过 40 个 Gabor 小波滤波后的输出形成的集合为：

$$S = \{O_{\nu,\mu}(\vec{z}) : \nu \in \{0,\cdots,4\}, \mu \in \{0,\cdots,7\}\}$$

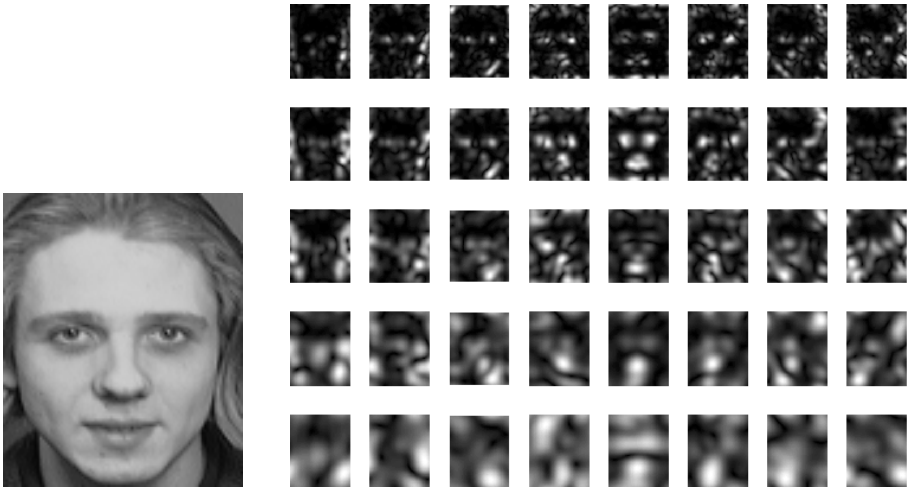
应用卷积定理，能够通过快速傅里叶变换计算 $O_{\nu,\mu}(\vec{z})$：

$$\Im\{O_{\nu,\mu}(\vec{z})\} = \Im\{I(\vec{z})\}\Im\{\psi_{\nu,\mu}(\vec{z})\} \tag{4.3.6}$$

$$O_{\nu,\mu}(\vec{z}) = \Im^{-1}\{\Im\{I(\vec{z})\}\Im\{\psi_{\nu,\mu}(\vec{z})\}\} \tag{4.3.7}$$

式中，符号 $\Im$ 和 $\Im^{-1}$ 分别定义了 FFT 和 IFFT。

图 4.5 显示了某幅图像经过 40 个 Gabor 小波滤波后取幅度值所形成的图像。图 4.5 和图 4.3 中的小波核函数是一一对应的，从直观来看，图 4.5（b）中第一行的图像分辨率最高，以后各行的图像分辨率逐行递减。这是因为第一行的图像尺度标称 $\nu = 0$，相应的 k_ν 最大，所以 Gabor 核函数的高斯采样窗小，采样频率大，对应滤波图像的分辨率最高；而最后一行的 $\nu = 4$，高斯采样窗大，对应的滤波图像分辨率最低。



（a） 原始图像　　（b） Gabor 滤波图像的幅度值

图 4.5　Gabor 小波变换示意图

由式（4.3.5）和图 4.5 可知，一幅图像经过 40 个不同幅度和方向的 Gabor 小波滤波后，形成 40 个尺寸相同的滤波图像，对于原始图像上任意一点 $\vec{z}_i = (x_i, y_i)$，把滤波图像上相同位置的点的幅度值串接起来，形成一个 40 维的

列向量，我们把这个列向量定义为“结”，用 $J(\vec{z}_i)$ 表示，即：

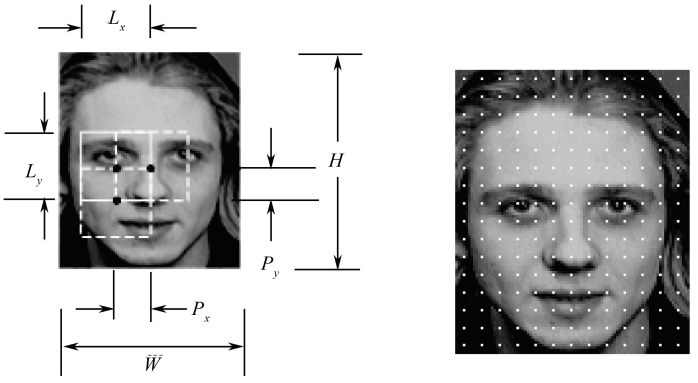
$$J(\vec{z}_i)=\{J_{\nu,\mu}(\vec{z}_i)=\left|O_{\nu,\mu}(\vec{z})\right|_{\vec{z}=\vec{z}_i} \quad \nu\in\{0,\cdots,4\},\mu\in\{0,\cdots,7\}\} \tag{4.3.8}$$

在文献[1]中利用弹性图来表示这些结之间的关联。本节在进行特征提取后，将使用 HMM 表示这些结之间的关联，并使用图 4.6 所示的方法对图像进行均匀采样。假设人脸图像的宽度为 W，高度为 H，用宽度为 L_x、高度为 L_y 的滑动采样窗（$L_x\times L_y$）对图像从上到下、从左到右进行采样，采样窗水平方向的重叠为 P_x，垂直方向的重叠为 P_y。采样时只把采样窗口的中心点作为人脸图像的“结”，忽略窗口内的其他点，也就是说，一个采样窗对应一个“结”。采样个数由下式给出：

$$T_0=\left\lfloor\frac{H-L_y}{L_y-P_y}\right\rfloor+1 \tag{4.3.9}$$

$$T_1=\left\lfloor\frac{W-L_x}{L_x-P_x}\right\rfloor+1 \tag{4.3.10}$$

式中，T_0 表示垂直方向采样点的个数，T_1 表示水平方向采样点的个数，符号$\lfloor\ \rfloor$表示向下取整（如$\lfloor 5.6\rfloor=5$）。本节中采样窗口的大小为7×7、9×9、11×11。重叠为 $P_x=0$，$P_y=0$。



（a）滑动窗口示意图　　（b）均匀采样示意图

图 4.6　均匀采样方案

需要说明的是以 Nefian 为代表的学者，在利用 HMM 进行人脸识别时，通常也采用上述方案进行特征提取，但是他们是对大小为 $L_x\times L_y$ 的采样窗口进行 2D-DCT，所以重叠 P_x 和 P_y 的取值影响了采样窗口的个数，进而对识别率产生影

响。而本节仅利用采样窗口的中心点，只有相邻两点之间的距离 L_x 和 L_y 影响采样点的个数，重叠 P_x 和 P_y 对采样个数没有影响，之所以使用它们是便于和 Nefian 等人的方法进行比较。

4.3.4　主元分析降维

由式（4.3.8）可知，$J(\vec{z}_i)\in R^D$，其中 $D=40$。文献[24]指出人脑总是在低维空间中完成诸如相似性判别的感知任务，因此低维对于机器学习同样非常重要。此外，维数的降低也减少了下一步使用 HMM 对人脸图像进行训练的计算量。每个结之间存在相关性，这将影响系统的识别性能。主元分析（Principal Component Analysis，PCA）法是一种良好的去相关方法[25]，是在最小均方差准则下把高维数据投影到低维空间的最优方法。假设共有 R 个训练图像，则所有结点的均值为：

$$\mu_J=\frac{1}{RT_0T_1}\sum_{r=1}^{R}\sum_{i=1}^{T_0T_1}J^{(r)}(\vec{z}_i) \tag{4.3.11}$$

对每个结进行中心化得：

$$\tilde{J}(\vec{z}_i)=J(\vec{z}_i)-\mu_J \tag{4.3.12}$$

由式（4.3.12）可以得出 $\sum_{r=1}^{R}\sum_{i=1}^{T_0T_1}\tilde{J}^{(r)}(\vec{z}_i)=0$。向量 $\tilde{J}(\vec{z}_i)$ 的协方差为：

$$\Sigma_J=\frac{1}{RT_0T_1}\sum_{r=1}^{R}\sum_{i=1}^{T_0T_1}\tilde{J}^{(r)}(\vec{z}_i)\left(\tilde{J}^{(r)}(\vec{z}_i)\right)^{\mathrm{T}} \tag{4.3.13}$$

式中，符号 T 表示向量或矩阵的转置，$\Sigma_J\in R^{D\times D}$。由线性代数理论可知协方差矩阵 Σ_J 可分解为：

$$\begin{aligned}&\Sigma_J=\Phi\Lambda\Phi^{\mathrm{T}}\\&\Phi=(\phi_1,\phi_2,\cdots,\phi_D)\\&\Lambda=\mathrm{diag}(\lambda_1,\lambda_2,\cdots,\lambda_D)\end{aligned} \tag{4.3.14}$$

式中，$\Phi\in R^{D\times D}$ 是 Σ_J 的特征向量所形成的矩阵，$\phi_i\in R^D$ 是 Σ_J 的特征向量。$\Lambda\in R^{D\times D}$ 是对角矩阵，主对角线元素是 Σ_J 的特征值，与特征向量一一对应，且 $\lambda_1\geqslant\lambda_2\geqslant\cdots\geqslant\lambda_D$。

在最小均方差准则下，PCA 可以使用部分主元最佳地重建信号，这样既可以对信号进行降维，又可以达到去相关的目的。设 $\Upsilon=(\phi_1,\phi_2,\cdots,\phi_d)$，$d\leqslant D$，则 $\Upsilon\in R^{D\times d}$。降维表达式为：

$$J_Y(\vec{z}_i) = Y^{\mathrm{T}} \tilde{J}(\vec{z}_i) \tag{4.3.15}$$

低维向量 $J_Y(\vec{z}_i) \in R^d$ 表达了原始向量 $\tilde{J}(\vec{z}_i)$ 的本质属性，我们把经过变换后的结 $J_Y(\vec{z}_i)$ 定义为“特征结”，并把它作为下一步 HMM 的观测向量。把包含“特征结”的人脸图像定义为“Gabor 特征脸”，简称“Gabor 脸”。

4.3.5 HMM 人脸识别

Samaria[22]在最初提出人脸的 HMM 时是基于这样一种考虑的：人脸图像包含头发、额头、眼睛、鼻子和嘴巴 5 个显著的特征区域，即使头部有一些偏转或倾斜，它们的次序从上到下保持不变，可以认为这 5 个显著区域隐含 5 个状态。事实上，在求 HMM 的初始参数时，我们可以进行类似的均匀分割，但是最终求得的状态是不受这些区域限制的，而且也不一定非要把人脸分成 5 个状态。因而，可以更进一步理解，我们观测到的序列是由若干个状态产生的，这些状态是抽象的，不具有具体的意义，只能通过观测序列对它进行估计。

笔者认为 HMM 的本质是基于统计分布一致性的聚类分析，每个隐含的状态就是一个聚类，对 HMM 进行训练的过程就是寻找每个聚类之间统计关联的过程。转移矩阵 A 表示了聚类之间的关联，而每个聚类的性质由概率分布矩阵 B 来决定，并通过观测序列 O 来表现。因此，可以利用 HMM 来表示“特征结”之间的相互关联，并把它用于人脸识别。本书采用如图 4.1 所示的左右型 HMM 是基于如下事实：对于人脸图像的某一部分，与其统计关联较强的是其邻域部分，而与其物理位置较远的部分统计关联弱，左右模型可以近似表达上述思想。

HMM 训练就是要为每个类别确定一组经过优化的 HMM 参数，每个模型可以用单幅或多幅图像进行训练，训练步骤如下。

① 对人脸进行 Gabor 变换，求出特征结，并将其作为观测向量，即 $o_i = J_Y(\vec{z}_i)$。

② 建立一个通用的模型 $\lambda = (A, B, \Pi)$，确定模型的状态数、允许的状态转移和高斯混合概率成分的个数。

③ 将训练数据均匀分割，与 N 个状态对应，计算模型的初始参数。对于状态转移矩阵 $A = \{a_{ij}\}$，可以使 $a_{ij} = 0$（$j < i$ 或 $j > i+1$）。对于初始概率分布 $\Pi = \{\pi_i\}$，可以使 $\pi_1 = 0$，$\pi_i = 0 (i \neq 1)$，即 HMM 从第一个状态开始。

概率分布矩阵 $B = \{b_j(o_i)\}$ 可依据下面的公式计算：

$$b_j(o_i) = \sum_{k=1}^{M} c_{jk} (2\pi)^{-d/2} \left|\Xi_{jk}\right|^{-1/2} \exp[-(o_i - \hat{\mu}_{jk})^T \Xi_{jk}^{-1} (o_i - \hat{\mu}_{jk})/2] \tag{4.3.16}$$

式中，c_{jk} 为混合成分的比例因子，即用高斯混合模型对概率分布矩阵 B 建模。$\hat{\mu}_{jk}$ 和 Ξ_{jk} 分别为高斯混合模型的均值和协方差矩阵：

$$\hat{\mu}_{jk} = (\sum_{i=1}^{E_{j,k}} o_i^{(j,k)}) / E_{j,k} \tag{4.3.17}$$

$$\Xi_{jk} = [\sum_{i=1}^{E_{j,k}} (o_i^{(j,k)} - \hat{\mu}_{jk})(o_i^{(j,k)} - \hat{\mu}_{jk})^T] / E_{j,k} \tag{4.3.18}$$

式中，$E_{j,k}$ 是均匀分割后，状态 j 的第 k 个混合成分对应的序列长度；$o_i^{(j,k)}$ 为相应的观测向量。

④ 用 Viterbi 分割取代均匀分割，并利用分段 K 均值聚类方法[26]求出高斯混合模型的参数，迭代调整初始模型参数。

⑤ 采用 Baum-Welch 算法对参数进行重新估计，并用最后得到的模型优化参数表示人脸数据库中的某个类别。

Viterbi 分割与 Baum-Welch 参数估计都需要迭代求解，本书利用对数评估值设定收敛条件。对于 Viterbi 分割的对数评估值为 $\Theta = \log P(O, Q^* \mid \lambda)$，参数估计的对数评估值为 $\Theta = \log P(O \mid \lambda)$，当

$$\frac{|\Theta - \Theta'|}{(\Theta + |\Theta'|)/2} < \varepsilon \tag{4.3.19}$$

时，则认为收敛，其中 Θ' 为前一次迭代的对数评估值，$\varepsilon = 10^{-4}$。图 4.7 显示了迭代次数与对数评估值之间的关系。从图中可以看出，对于 Viterbi 分割，当迭代次数大于 10 次时，基本收敛；对于参数估计，当迭代次数大于 20 次时，基本收敛。所以我们在实验中利用式（4.3.19）或最大迭代次数来判定是否收敛，只要满足其中之一，就认为已经收敛，停止迭代。图 4.8 给出了 HMM 训练流程图。

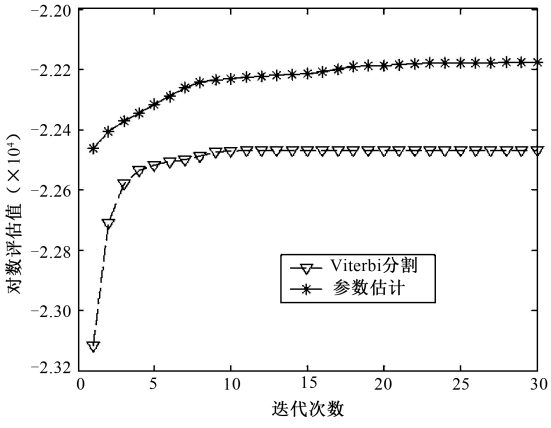


图 4.7　迭代次数与对数评估值之间的关系

在人脸识别阶段，首先要对待识别的人脸图像 k 进行 Gabor 变换，计算它的特征结，形成观测序列 $O^{(k)}$，然后使用前向-后向算法计算每个训练模型 λ_i 产生该序列的概率 $P(O^{(k)} \mid \lambda_i)$，最大值所对应的模型就是待识别人脸图像所属的类别，可以用公式表达为：

$$\lambda_n = \arg \max_i P(O^{(k)} \mid \lambda_i) \tag{4.3.20}$$

即如果第 n 个模型 λ_n 产生序列 $O^{(k)}$ 的概率最大，则将图像 k 归入第 n 类。图 4.9 给出了 HMM 人脸识别流程图。

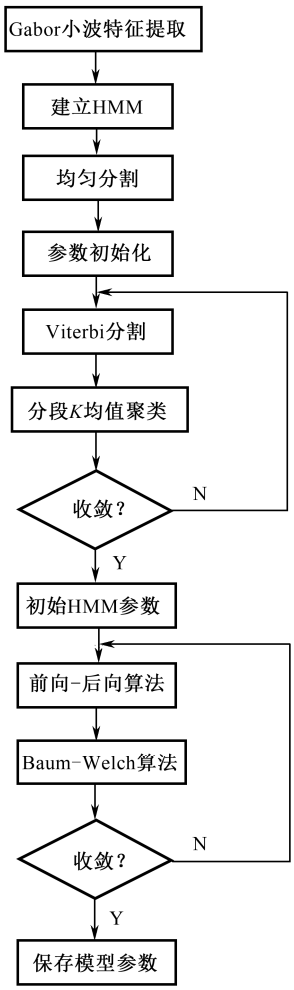


图 4.8　HMM 训练流程图

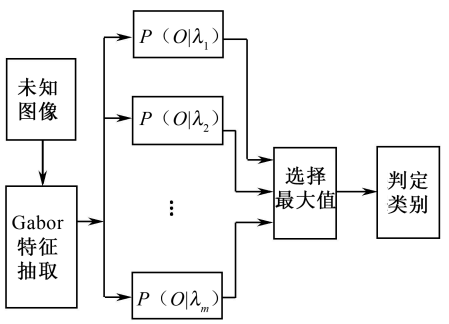


图 4.9　HMM 人脸识别流程图

4.3.6　算法复杂度分析

HMM 算法的复杂度一般指计算隐含层中 $P(O|\lambda)$ 所需的乘法和加法的总和，为了便于与 Nefian 等人的方法进行比较，全面衡量整个系统的复杂度，本书主要分析观测层和 Gabor 特征抽取的计算时间复杂度。由于乘法运算所需时间更长，所以本书只分析这两个过程所需的乘法数量。

1．观测层复杂度

这主要指计算式（4.3.16）指数部分 $(o_i-\hat{\mu}_{jk})^T \Xi_{jk}^{-1}(o_i-\hat{\mu}_{jk})$ 所需的计算量，向量的维数为 d，混合成分的个数为 M，所以复杂度为：

$$C_{\mathrm{OL}} \approx d^2 NMT_1T_0 \tag{4.3.21}$$

2．特征提取复杂度

特征提取复杂度为 Gabor 变换的计算量和 PCA 的计算量之和。对于 Gabor 变换，由式（4.3.5）实现，涉及计算二维 FFT[27]。其复杂度为：

$$C_{\mathrm{G}} = DWH(\log_2 WH) + DWH + WH(\log_2 WH)/2$$

式中，D 为 Gabor 小波的个数，由于 $D \gg 1$，$\log_2 WH \gg 1$，所以

$$C_{\mathrm{G}} \approx DWH(\log_2 WH) \tag{4.3.22}$$

对于 PCA[27]，主要涉及式（4.3.13）～式（4.3.15）的计算，其复杂度为：

$$C_{\mathrm{P}} \approx T_1T_0D^2 + D^3/m + T_1T_0Dd$$

式中，m 为训练图像的总数，由于 $T_1T_0 \gg 1$，$D \gg d$，所以

$$C_{\mathrm{P}} \approx T_1T_0D^2 \tag{4.3.23}$$

由式（4.3.21）和式（4.3.22）可知，特征提取所需的复杂度为：

$$C_{\mathrm{F}} \approx DWH(\log_2 WH) + T_1T_0D^2 \tag{4.3.24}$$

由文献[9]可知 1D-HMM 隐含层的计算复杂度为

$$C_{\mathrm{HL}} = N^2T_1T_0 \tag{4.3.25}$$

假设使用 FFT[28]计算 2D-DCT，则对于 Nefian 等使用 2D-DCT 进行特征提取所需的复杂度为：

$$C_{\mathrm{F}} \approx 3L_xL_y(\log_2 L_xL_y)T_1T_0/8 \tag{4.3.26}$$

表 4.1 汇总了与复杂度相关的符号含义。表 4.2 给出了相关方法的复杂度对比。表 4.3 给出了在 ORL 人脸数据上相关算法复杂度的具体比较。其中，N_0 和 N_1 是伪 2D-HMM 中垂直和水平状态的个数。从表中可以看出，由于 Samaria 直接取图像亮度作为观测向量，所以特征提取的复杂度几乎为 0，但是由于观测向

量的维数高，观测向量的个数多，所以观测层和隐含层的计算量都很大，识别率也不高。Nefian 采用 2D-DCT 系数作为观测向量，其特征提取的复杂度低，对于 1D-HMM，其观测层和隐含层的复杂度也较低，但是识别率仅为 86%；对于 EHMM，尽管它的识别率很高，但是采用的是伪 2D-HMM，观测向量的个数多，状态总数多，所以观测层和隐含层的复杂度也较高。本书的方法在特征提取上所需的计算量较大，但观测向量的个数较少，状态数也少，所以观测层和隐含层的复杂度低，而且仅使用 1D-HMM 识别率达到 99%，因此整体性能较其他方法好。

表 4.1　汇总了与复杂度相关的符号含义

符号	含　义	符号	含　义
d	观测向量维数	N	状态总数
D	Gabor 小波个数	L_x	窗口宽度
M	混合成分的个数	L_y	窗口高度
T_0	垂直方向观测向量个数	P_x	窗口水平重叠
T_1	水平方向观测向量个数	P_y	窗口垂直重叠
N_0	垂直方向状态个数	W	图像宽度
N_1	水平方向状态个数	L	图像高度
C_{OL}	观测层复杂度	C_{HL}	隐含层复杂度
C_{F}	特征提取复杂度	m	图像类别总数
×	未使用某项指标	R	训练图像总数

表 4.2　相关方法的复杂度对比

方　法	识别率（%）	特征提取复杂度	观测层复杂度	隐含层复杂度
1D_HMM_Lum[2]	84.0	≈ 0	$d^2N_0MT_1T_0$	$N_0^2T_0$
1D_HMM_DCT[3]	86.0	$3L_xL_y(\log_2 L_xL_y)T_1T_0/8$	$d^2N_0MT_1T_0$	$N_0^2T_0$
2D_PHMM_Lum[2]	94.5	≈ 0	$d^2NMT_1T_0$	$(\sum_{k=1}^{N_0}N_1^k)^2T_1T_0$
2D_EHMM_DCT[3]	99.0	$3L_xL_y(\log_2 L_xL_y)T_1T_0/8$	$d^2NMT_1T_0$	$(\sum_{k=1}^{N_0}(N_1^k)^2)T_1T_0+N_0^2T_0$
本书方法	99.0	$DWH(\log_2 WH)+T_1T_0D^2$	$d^2NMT_1T_0$	$N^2T_1T_0$

表 4.3　ORL 人脸数据上相关算法复杂度的具体比较

方 法	1D_HMM_Lum	1D_HMM_DCT	2D_PHMM_Lum	2D_EHMM_DCT	本书方法
识别率	84.0	86.0	94.5	99.0	99.0
N_0	5	5	5	5	×
N_1	1	1	3，6，6，6，3	3，6，6，6，3	×
N	5	5	24	24	5
L_x	92	92	8	8	7
L_y	10	10	10	10	7
P_x	×	×	6	6	0
P_y	9	8	8	8	0
T_0	103	52	52	52	16
T_1	1	1	43	43	13
T_1 T_0	103	52	2236	2236	208
d	920	39	80	6	6
M	1	1	1	3	3
C_{F}	0	176628	0	424075	5827271
C_{OL}	435896000	395460	343449600	5795712	112320
C_{HL}	2575	1300	1287936	283036	5200

4.3.7　实验结果及分析

本书使用 ORL 人脸数据库，该数据库包含 40 个人，每人 10 幅图像，共 400 幅图像，每个图像大小为$W=92$，$L=112$。实验中每人取 5 幅图像，共 200 幅图像进行训练，用另外 200 幅进行识别。第一种实验是对待识别图像直接进行识别；第二种实验是先对待识别图像手工进行部分遮挡，然后再进行识别。

1．非遮挡图像的识别性能

为了考察 Gabor 脸的分类性能，我们首先单独使用它进行人脸识别，在判决中使用与统计模型相关的 Mahalanobis 距离，采用最近邻法进行判决：

$$n=\arg\min_{j}\sum_{i=1}^{T_1T_0}\left((J_Y^{(k)}(\vec{z}_i)-J_Y^{(j)}(\vec{z}_i))^T\Sigma_J^{-1}(J_Y^{(k)}(\vec{z}_i)-J_Y^{(j)}(\vec{z}_i))\right) \tag{4.3.27}$$

式中，$\Sigma_J \in R^{D\times D}$ 为所有训练图像特征结的协方差矩阵，即如果待识别图像 k 与第 n 个训练图像的距离最小，则将待识别图像 k 归入第 n 个训练图像所属于的类别。图 4.10 显示了 Gabor 脸与单独使用 DCT 在相同的窗口下识别性能的对比。Gabor 特征脸的识别性能最好，最高识别率为 95.5%；其次是单独使用 Gabor 小波；而 DCT 的分类性能最差，最高识别率只有 75%。可见图像经过 Gabor 小波处理后，包含了更多的判决信息。此外，单独使用 Gabor 小波识别性能也不高，但是经过 PCA 处理后，识别率明显增高，说明经过 Gabor 变换后，再进行 PCA 处理是十分必要的，PCA 有效去除了向量之间的相关性。

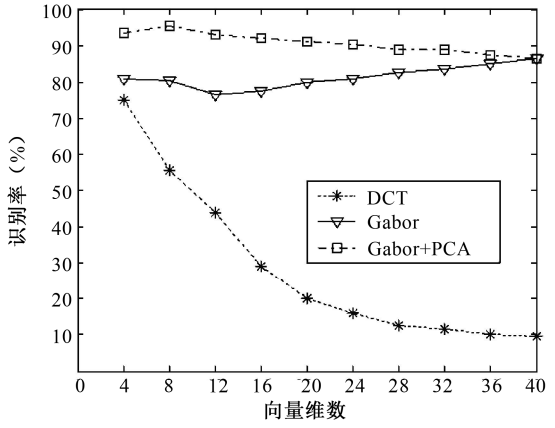


图 4.10　Gabor 脸的识别性能

图 4.11 显示了 Gabor 脸与 HMM 相结合的识别性能。从图中可见，当观测向量维数较小时，利用 $J_Y(\vec{z}_i)$ 作为 HMM 的观测向量（对应 Gabor+PCA+HMM）的识别性能最好，识别率达到 99%。而直接用 $J(\vec{z}_i)$ 作为 HMM 的观测向量（对应 Gabor+HMM）的识别性能较差，这又一次说明利用 PCA 进行去相关和降维是非常必要的。当维数 $d \in \{6,\cdots,12\}$ 时，Gabor 小波的尺度小，高斯窗采样频率大，识别率较高；在此范围外，识别率递减。这说明相对较小尺度的 Gabor 变换包含较多的人脸特征，观测向量之间的相关性较弱，对识别率的改善作用较大。但是这并不等于大尺度的 Gabor 变换没有用处，实际上高频信息描述了图像的局部细节，低频信息描述了图像的全局特征，而 PCA 的作用是把两种信息进行有效的融合，去除相关性，提高识别率。

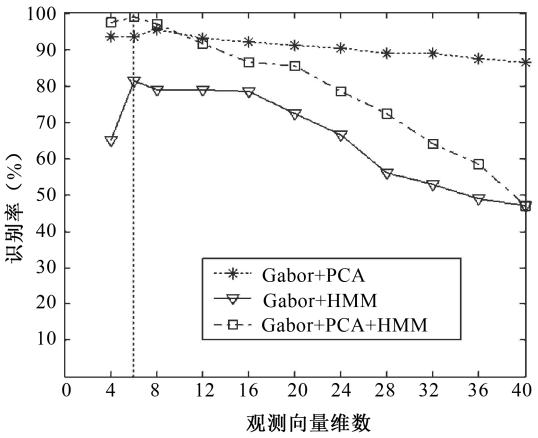


图 4.11　Gabor 脸与 HMM 相结合的识别性能

图 4.12 显示了 7×7 窗口下观测向量维数与识别率的关系。图中的图例表示 HMM 状态数和高斯概率混合成分个数。例如，5×3 表示本条曲线是在状态数 $N=5$，混合成分个数 $M=3$ 的条件下绘制的。从图中可见，当观测向量维数 $d\in\{6,\cdots,12\}$ 时，识别率很高，而且在 $d=6$ 附近识别率最高。当 $d>12$ 时，随着维数的增高，识别率递减；当 $d<6$ 时，随着维数的增高，识别率递增。这是因为维数太低，包含的人脸特征也少，所以识别率降低。而维数过高，一方面由于 PCA 只利用了图像的二阶统计信息，不可能完全去相关，所以随着维数的增加，观测向量之间的相关性增大，干扰了 HMM 训练图像的能力，因此识别率降低；另一方面，图像的本质特征总是包含在低维空间中[8]，所以观测向量的维数越高，识别率越低。

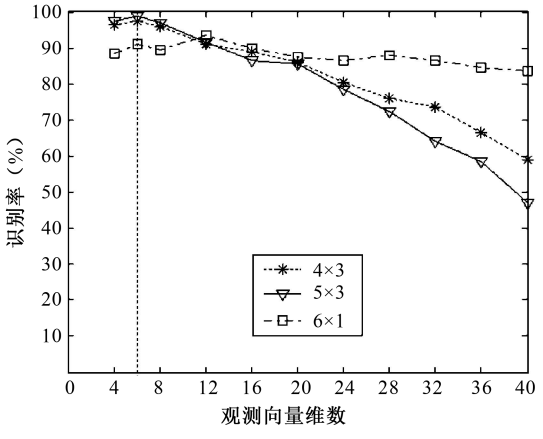


图 4.12　7×7 窗口下观测向量维数与识别率的关系

图 4.13 显示了不同采样窗口下的识别性能比较。图例中的第一部分表示采样窗口的大小，第二部分表示状态数和高斯概率混合成分个数。由于采样滑动窗口越小，观测序列的长度越大，HMM 将拥有更多的训练数据，所以整体识别率增高。例如，7×7 窗口下的整体识别率最高，但是窗口越小，观测序列的长度越大，训练 HMM 的时间越长，因此窗口的选择也要适当。在 $d=6$ 附近，不同采样窗口的识别率都达到最大值，这又一次说明图像的本质特征总是包含在低维空间中。

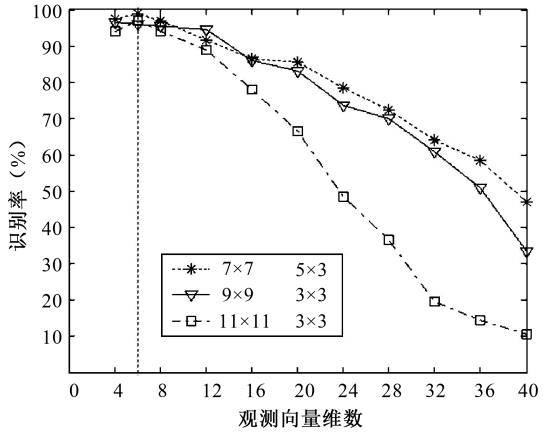


图 4.13　不同采样窗口下的识别性能比较

表 4.4 显示了在 7×7 窗口下，观测向量维数 $d=6$ 时，状态个数 N 和高斯概率混合成分个数 M 对识别率的影响。从表中可见，随着 M 增大，识别性能得到改善，但当 M 增大到一定量时，此时若再增大 M，则识别率降低。由于用高斯混合模型描述概率分布矩阵 B，而矩阵 B 描述了每个状态的概率统计属性，适当增大 M 可以增加高斯混合模型的自由度，因此可以更好地表现每个状态的统计属性，进而识别率增高。但是如果 M 增大过量，就使高斯混合模型过分依赖训练数据，对当前训练数据的描述过于精确，以至于测试数据与训练数据稍有不同时，就无法识别出测试数据，这实际上是过学习的结果。

表 4.4　状态个数和高斯概率混合成分个数对识别率的影响

状态个数	高斯概率混合成分个数					
	1	3	5	7	9	11
3	83.5	95.0	96.5	95.0	94.5	94.0
4	85.5	97.5	95.5	95.0	93.0	93.0
5	90.0	99.0	97.5	96.5	93.5	91.5

续表

状态个数	高斯概率混合成分个数					
	1	3	5	7	9	11
6	91.0	97.5	96.5	93.0	90.5	89.5
7	91.5	96.5	95.5	93.0	91.0	86.0
8	93.5	96.5	93.5	90.5	85.0	84.0
9	94.5	98.0	94.0	90.5	85.0	73.5
10	94.5	96.0	91.0	86.0	78.0	71.5

从表 4.4 中还可看出，为了达到更好的识别效果，当 N 较大时，为了避免过学习，使模型不过分依赖训练数据，就要使 M 稍小，如 $N=5$，$M=3$；当 N 较小时，为了更充分地描述模型，就要使 M 稍大，如 $N=3$，$M=5$。同理，在其他窗口中也有此规律。从聚类分析的角度看，若把每个状态看成一个聚类，在观测数据一定的条件下，聚类数 N 小，每个聚类包含的数据就多，就需要使用更多的参数来描述这个聚类，所以相应的 M 大；若聚类数 N 大，每个聚类包含的数据就少，相应的 M 小。因此可以把 HMM 看成是基于统计分布一致性的聚类分析，每个隐含的状态就是一个聚类，对 HMM 进行训练的过程就是寻找每个聚类之间统计关联的过程。转移矩阵 A 表示了聚类之间的关联，而每个聚类的性质由概率分布矩阵 B 来决定，并通过观测序列 O 来表现。

2．部分遮挡图像的识别性能

若手工对 200 幅待识别的测试图像进行部分遮挡，本书方法仍能保持较高的识别率。表 4.5 对相关方法遮挡前后的识别率进行了对比。表 4.3 显示了相关方法所用的具体参数。

表 4.5　相关方法遮挡前后的识别率对比

方　法	R_1（%）	R_2（%）	R_1-R_2（%）
2D_EHMM_DCT[3]	99.0	76.0	23.0
DCT	75.0	69.5	9.5
1D_HMM_DCT[3]	86.0	55.5	30.5
Gabor	86.5	83.0	3.5
Gabor+PCA	95.5	94.5	1.0
本书方法	99.0	98.5	0.5

注：R_1 指遮挡前识别率，R_2 指遮挡后识别率。

从表 4.5 可以看出，与 Gabor 相关的方法，遮挡前后识别率的变化较小；而采用 DCT 的方法，遮挡前后的识别率变化大。对这个结果的合理性解释如下。

（1）在特征提取时，本书把每个人脸图像与 40 个不同方向和不同尺度的 Gabor 滤波器进行卷积，然后在图像上放置一组网格结点，每个结点用该结点处的多尺度 Gabor 幅度特征描述，经过 PCA 变换后，形成特征结。可见每个特征结浓缩了不同方向和不同尺度的多分辨率信息，而且由于采用卷积操作，每个特征结的信息都是与图像的整体相关联的，当对图像进行部分遮挡时，部分特征结的信息损失了，但是其他特征结仍然包含图像的整体信息，对判决仍然具有贡献，所以识别性能较好。此外，小波变换的系数具有冗余度[29]，可以使用部分小波系数对图像进行重建，这种特性对图像识别的判决是非常有利的。而与 DCT 相关的方法是对每个采样窗进行 DCT 变换，形成观测向量，尽管采样窗之间有重叠，但是每个采样窗所关联的信息是局部信息，与图像整体信息的关联弱，当对图像进行部分遮挡时，被遮挡部分的局部信息丢失了，而从图像的其他部分几乎得不到遮挡部分的信息，因此遮挡后，识别率明显降低。尽管 EHMM 算法的观测序列的长度为 2236（对应采样窗的个数），本书算法观测序列的长度只有 208，但是由于本书采用 Gabor 脸进行特征提取，所以对图像的部分遮挡具有更大的容忍度，系统的整体性能更佳。

（2）在最小均方差准则下，PCA 可以使用部分主元最佳地重建信号。文献[30]指出，利用这种技术可以对部分遮挡的图像进行重建。本书在特征提取时也使用了 PCA，所以这在一定程度上增强了系统的识别性能。表 4.5 显示了单独使用 Gabor+PCA 的方法，识别性能也较佳，且遮挡前后识别率变化仅为 1%。

（3）HMM 可以有效地把每个特征结关联起来，这就使该算法对图像的部分遮挡具有更大的容忍度，有效改善了分类器的识别性能。

图 4.14 显示了人脸图像部分遮挡后，使用 EHMM 算法和本书算法进行识别的结果。从图 4.14 中可见，本书算法具有更好的识别性能。图 4.15 显示了部分遮挡的示意图，×表示误识。

（a）待识别图像

（b）EHMM 算法的识别结果

（c）本书算法的识别结果

图 4.14　图像遮挡后 EHMM 算法和本书算法进行识别的结果

图 4.15　部分遮挡的示意图

4.3.8　结论

本节提出了一种基于 Gabor 变换和 1D-HMM 的人脸识别方法，对算法复杂度进行了分析，并同 Samaria 和 Nefian 的方法进行了比较。实验结果表明，本书方法识别率高，复杂度较低，对图像的部分遮挡具有更大的容忍度。此外，本节还分析了观测向量维数与识别率的关系，以及状态个数和高斯概率混合成分个数对识别率的影响，定性描述了 HMM 的本质属性。

基于 HMM 的人脸识别方法具有以下优点：第一，允许人脸有表情变化和较大的头部转动；第二，有较高的识别率；第三，对部分遮挡的图像具有较大的容忍度。另外，训练模型可在建立数据库时完成，计算时间也是可以接受的，因此基于 HMM 的人脸识别方法具有较好的发展前景。

本书算法在特征提取上所需的计算量较大，在今后的研究中，可采用快速算法降低复杂度，使本书算法更有利于工程应用。

4.4　基于 Gabor 小波、ICA 和 HMM 的人脸识别方法

前面我们采用主元分析法对经过 Gabor 小波变换所形成的每个结点进行去相关、降维，取得了较好的识别结果。本节我们使用独立元分析法对每个结点进行去相关，也能获得较好的结果。

4.4.1　独立元分析降维

独立元分析[31]（Independent Component Analysis，ICA）法是一种良好的去相关方法，它利用数据的高阶统计量，获取数据的独立成分，从而提取随机数据集的本质特征。ICA 是在最小化随机数据集各成分之间统计相关性的基础上，寻找一个线性变换 F，并利用此变换把高维数据投影到低维空间，进而达到去相关和降维的目的。

首先根据前面所述的方法，对人脸图像进行多分辨率的 Gabor 小波变换，然后在图像上放置一组网格结点，接着采用独立元分析法对每个结点进行去相关、降维。为了方便起见，我们把式（4.3.8）所定义的 $J(\vec{z}_i)$ 看成一个随机向量，则 $J(\vec{z}_i)$

的协方差矩阵为：

$$\Sigma_{\mathfrak{Z}} = E((J(\vec{z}_i) - E(J(\vec{z}_i)))(J(\vec{z}_i) - E(J(\vec{z}_i)))^{\mathrm{T}}) \tag{4.4.1}$$

式中，$E(\cdot)$ 表示求随机向量的期望，T 表示向量或矩阵的转置，$\Sigma_{\mathfrak{Z}} \in R^{D\times D}$。利用 ICA，可把协方差矩阵分解为：

$$\Sigma_{\mathfrak{Z}} = F\Delta F^{\mathrm{T}} \tag{4.4.2}$$

式中，$\Delta \in R^{d\times d}$，是一个对角矩阵，主对角线上的元素为正实数；$F \in R^{D\times d}$。可把原始随机向量 $J(\vec{z}_i)$ 变换成低维向量 $J_F(\vec{z}_i) \in R^d$：

$$J(\vec{z}_i) = FJ_F(\vec{z}_i) \tag{4.4.3}$$

新的低维向量 $J_F(\vec{z}_i)$ 各分量之间是相互独立的，可见解决问题的关键是求出变换矩阵 F。

设 $p_F(\vec{u})$ 为随机向量 $J_F(\vec{z}_i)$ 的概率密度函数，向量 $J_F(\vec{z}_i)$ 各分量之间相互独立的充要条件是：联合概率密度等于边缘概率密度的积，即

$$p_F(\vec{u}) = \prod_{i=1}^{d} p_{F_i}(u_i) \tag{4.4.4}$$

为了求出变换矩阵 F，Comon[32]提出了一种用于度量随机向量各分量之间独立性的优化准则，即最小化互信息准则。这个准则首先需要计算式（4.4.4）中两个概率密度函数之间的互信息：

$$I(p_F) = \int p_F(\vec{u}) \log \frac{p_F(\vec{u})}{\prod p_{F_i}(u_i)} \mathrm{d}\vec{u} \tag{4.4.5}$$

式（4.4.5）确定了向量 $J_F(\vec{z}_i)$ 的平均互信息。式（4.4.4）和式（4.4.5）表明，向量 $J_F(\vec{z}_i)$ 各分量之间相互独立的充要条件是互信息为 0。

由文献[32]可知，式（4.4.5）可写成如下形式：

$$I(p_F) = J(p_F) - \sum J(p_{F_i}) + \frac{1}{2}\log \frac{\Pi V_{ii}}{|V|} \tag{4.4.6}$$

式中，V 为向量 $J_F(\vec{z}_i)$ 的协方差矩阵。$J(p_F)$ 为负熵，用于度量 $p_F(\vec{u})$ 与高斯联合概率密度函数 $\phi_F(\vec{u})$ 之间的相似性：

$$J(p_F) = -\int p_F(\vec{u}) \log \frac{\phi_F(\vec{u})}{p_F(\vec{u})} \mathrm{d}\vec{u} \tag{4.4.7}$$

式（4.4.6）和式（4.4.7）提供了一种近似计算互信息的方法，根据上面的公式，文献[32]采用了一种最小化互信息的优化准则，从而求出变换矩阵 F，主要步骤如下：

① 首先对随机向量 $J(\vec{z}_i)$ 进行白化变换，去除各分量之间的相关性，并使协方差矩阵 $\Sigma_{\mathfrak{Z}}$ 为单位矩阵。

② 利用高阶累积量（κ-statistics）对白化后的数据进行一系列旋转变换，最小化式（4.4.6）右边的第二项，同时保持其他项不变，从而求出初始的变换矩阵 $\tilde{F}$ 。

③ 对初始的变换矩阵 $\tilde{F}$ 中的每列进行标准化，最后求出变换矩阵 F 。

关于 ICA 的详细计算步骤，参见文献[32]。低维向量 $J_F(\vec{z}_i) \in R^d$ 表达了原始向量 $J(\vec{z}_i)$ 的本质属性，把经过变换后的结 $J_F(\vec{z}_i)$ 定义为“特征结”，并把它作为下一步 HMM 的观测向量。

在 HMM 训练阶段，首先对人脸进行 Gabor 变换，结合 ICA 求出特征结，并将其作为观测向量，即 $o_i = J_F(\vec{z}_i)$ 。接下来的步骤与 4.3.5 节所述的内容基本相同，这里就不赘述了。

4.4.2　实验结果及分析

这里仍使用 ORL 人脸数据库，实验中每人取 5 幅图像，共 200 幅图像进行训练，用另外 200 幅进行识别。

为了考察 Gabor 小波的分类性能，首先单独使用它进行人脸识别，在判决中使用与统计模型相关的 Mahalanobis 距离，采用最近邻法进行判决：

$$n = \arg \min_j \sum_{i=1}^{T_1 T_0} \left((J_F^{(k)}(\vec{z}_i) - J_F^{(j)}(\vec{z}_i))^T V^{-1} (J_F^{(k)}(\vec{z}_i) - J_F^{(j)}(\vec{z}_i)) \right) \qquad (4.4.8)$$

式中，V 为所有训练图像特征结的协方差矩阵。如果待识别图像 k 与第 n 个训练图像的距离最小，则将待识别图像 k 归入第 n 个训练图像所属于的类别。图 4.16 显示了 Gabor 小波与单独使用二维离散余弦变换[3]（2D-DCT）在相同的窗口下识别性能的对比。可以看出，Gabor 小波与 ICA 结合形成特征结后的（对应 Gabor+ICA）识别性能最好，最高识别率为 92%；其次是单独使用 Gabor 小波；而 DCT 的分类性能最差，最高识别率只有 75%。所以使用 Gabor 小波提取特征要优于文献[3]所述的利用 DCT 提取特征，即图像经过 Gabor 小波处理后，包含了更多的判决信息。此外，单独使用 Gabor 小波识别性能也不高，但是经过 ICA 处理后，识别率明显增高，说明经过 Gabor 变换后，再进行 ICA 处理是十分必要的，ICA 有效去除了向量之间的相关性。

图 4.17 显示了 Gabor 特征结与 HMM 结合的识别性能。从图中可见，当观测向量维数较小（ $d < 16$ ）时，利用 $J_F(\vec{z}_i)$ 作为 HMM 的观测向量（对应 Gabor+ICA+HMM）的识别性能最好，最高识别率达到 99%。而直接用 $J(\vec{z}_i)$ 作为 HMM 的观测向量（对应 Gabor+HMM）的识别性能较差，这又一次说明利用

ICA 进行去相关和降维是非常必要的。对于 Gabor+HMM，当维数 $d \in \{4,\cdots,12\}$ 时，Gabor 小波的尺度小，高斯窗采样频率大，识别率较高；在此范围外，识别率递减。这说明相对较小尺度的 Gabor 变换包含较多的人脸特征，观测向量之间的相关性较弱，对识别率的改善作用较大。但是这并不等于大尺度的 Gabor 变换没有用处，实际上高频信息描述了图像的局部细节，低频信息描述了图像的全局特征，而 ICA 的作用是把两种信息进行有效的融合，去除相关性，提高识别率。

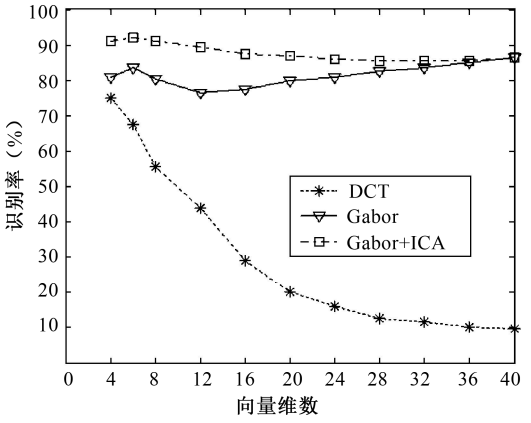


图 4.16　Gabor 小波的识别性能

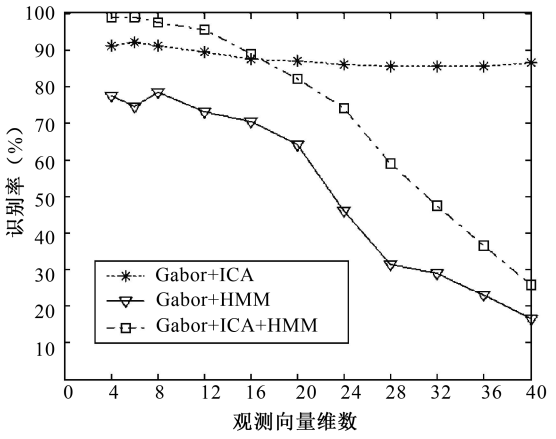


图 4.17　Gabor 特征结与 HMM 结合的识别性能

图 4.18 显示了 7×7 窗口下观测向量维数与识别率的关系。图中的图例表示 HMM 状态数和高斯概率混合成分个数。例如，5×3 表示本条曲线是在状态数 $N=5$，混合成分个数 $M=3$ 的条件下绘制的。从图中可见，当观测向量维数在

$d=8$ 附近时，识别率较高。当 $d>16$ 时，随着维数的增高，识别率递减；当 $d<6$ 时，随着维数的增高，识别率递增。这是因为维数太低，包含的人脸特征也少，所以识别率降低。而维数过高，一方面 ICA 所起的作用减弱，观测向量之间的相关性增强，干扰了 HMM 训练图像的能力，因此识别率降低；另一方面，图像的本质特征总是包含在低维空间中[24]，所以观测向量的维数越高，识别率越低。

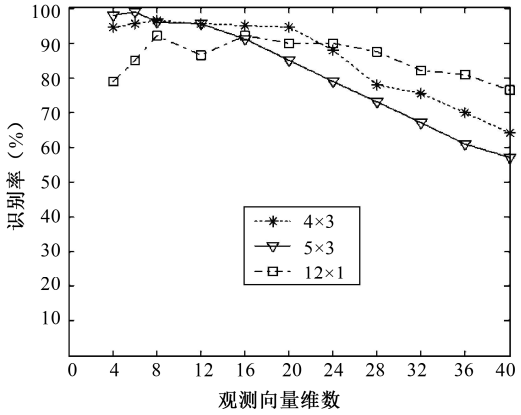


图 4.18　7×7 窗口下观测向量维数与识别率的关系

图 4.19 显示了不同采样窗口下的识别性能比较。图例中的第一部分表示采样窗口的大小，第二部分表示状态数和高斯概率混合成分个数。由于采样滑动窗口越小，观测序列的长度越大，HMM 将拥有更多的训练数据，所以整体识别率增高。例如，7×7 窗口下的整体识别率最高，但是窗口越小，观测序列的长度越大，训练 HMM 的时间越长，因此窗口的选择也要适当。在 $d=6$ 附近，不同采样窗口的识别率都较高，这又一次说明图像的本质特征总是包含在低维空间中。

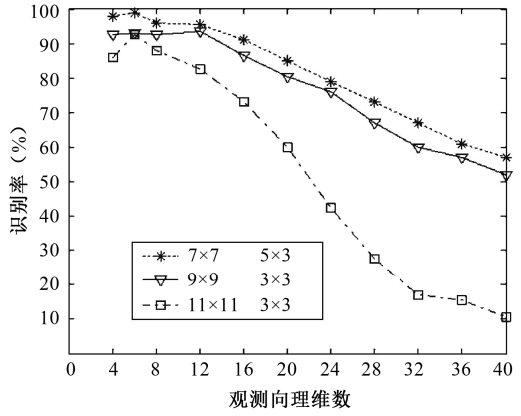


图 4.19　不同采样窗口下的识别性能比较

图 4.20 显示了在 7×7 窗口下，观测向量维数 $d=6$ 时，状态个数 N 和高斯概率混合成分个数 M 对识别率的影响。图中每条曲线都是在某个状态个数下绘制的。从图中可见，保持状态个数不变，随着 M 增大，识别性能得到改善；但当 M 增大到一定量时，此时若再增大 M，则识别率降低。由于用高斯混合模型描述概率分布矩阵 B，而矩阵 B 描述了每个状态的概率统计属性，适当地增大 M 可以增加高斯混合模型的自由度，因此可以更好地表现每个状态的统计属性，进而识别率增高。但是如果 M 增大过量，就使高斯混合模型过分依赖训练数据，对当前训练数据的描述过于精确，以至于测试数据与训练数据稍有不同，就无法识别出测试数据，这实际上是过学习的结果。

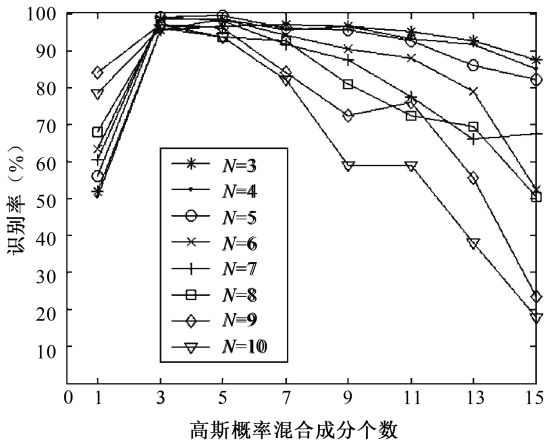


图 4.20　状态个数 N 和高斯概率混合成分个数 M 对识别率的影响

从图 4.20 中还可看出，为了达到更好的识别效果，当 N 较大时，为了避免过学习，使模型不过分依赖训练数据，就要使 M 稍小，如 $N=7$、$M=3$ 时，识别率为 99%；当 N 较小时，为了更充分地描述模型，就要使 M 稍大，如 $N=3$、$M=7$ 时，识别率为 97%。同理，在其他采样窗口中也有此规律。从聚类分析的角度看，若把每个状态看成一个聚类，在观测数据一定的条件下，聚类数 N 小，每个聚类包含的数据就多，就需要使用更多的参数来描述这个聚类，所以相应的 M 大；若聚类数 N 大，每个聚类包含的数据就少，相应的 M 小。因此可以把 HMM 看成是基于统计分布一致性的聚类分析，每个隐含的状态就是一个聚类，对 HMM 进行训练的过程就是寻找每个聚类之间统计关联的过程。转移矩阵 A 表示了聚类之间的关联，而每个聚类的性质由概率分布矩阵 B 来决定，并通过观测序列 O 来表现。

表 4.6 给出了在 ORL 人脸数据上相关算法识别率对比，其中的相关符号说

明见表 4.1。从表 4.6 中可以看出 Samaria[2]直接取图像亮度作为观测向量，观测向量的维数高，采用一维 HMM 识别率只有 84%。即使采用伪二维模型 PHMM，识别率也只有 94.5%，而且观测向量的维数高，观测向量的个数多。Nefian[3] 采用 2D-DCT 系数作为观测向量，降低了观测向量的维数，并取得了与 Samaria 相当的识别效果，识别率为 86%。尽管 Nefian 采用的 EHMM 具有较高的识别率，但是它实际上是伪 2D-HMM，观测向量的个数多，状态总数多，所以对模型进行训练所需的时间较长。本书方法的识别率达到 99%，而且仅采用 1D-HMM，所以观测向量的个数较少，状态数也少，因此整体性能较其他方法好。

表 4.6　ORL 人脸数据上相关算法识别率对比

方 法	1D_HMM_Lum	1D_HMM_DCT	2D_PHMM_Lum	2D_EHMM_DCT	本书方法
识别率	84.0	86.0	94.5	99.0	99.0
N_0	5	5	5	5	×
N_1	1	1	3，6，6，6，3	3，6，6，6，3	×
N	5	5	24	24	7
L_x	92	92	8	8	7
L_y	10	10	10	10	7
P_x	×	×	6	6	0
P_y	9	8	8	8	0
T_0	103	52	52	52	16
T_1	1	1	43	43	13
T_1 T_0	103	52	2236	2236	208
d	920	39	80	6	4
M	1	1	1	3	3

注：×表示未使用某项指标。

4.4.3　结论

在 4.3 节中，我们使用 PCA 方法进行去相关和降低维数，取得了较高的识别率；在本节中，我们用 ICA 方法同样取得了良好的识别效果。那么这两种方法有什么异同呢？

PCA 仅考虑了信号的二阶统计特性，也就是说，它仅利用了包含在协方差

矩阵中的信息，在最小均方差准则下，PCA 可以使用部分主元最佳地重建信号。因为高斯概率分布的所有三阶以上的累积量都是零[33]，所以当随机特征的概率分布呈高斯分布时，采用 PCA 进行特征抽取是比较合适的。ICA 充分利用了信号的高阶统计特性，能更好地减少信号之间的统计相关性，当随机特征的概率分布呈非高斯分布时，采用 ICA 进行特征抽取是比较合适的。人脸信息呈何种概率分布，目前尚无定论，对于本章所采用的具体算法来说，我们用混合高斯概率分布对 HMM 的概率分布矩阵 B 进行建模，因此用 PCA 进行去相关、降维看起来更适合于基于 HMM 的人脸识别算法。当把图 4.15 所示的遮挡图像作为测试图像时，用基于 ICA 的 HMM 人脸识别算法（以下简称 GICAH）进行识别，识别率为 81.5%，尽管比基于 EHMM 的算法识别率高（76%，见表 4.5），但要低于基于 PCA 的 HMM 人脸识别算法（以下简称 GPCAH），可见基于 PCA 的 HMM 人脸识别算法对部分遮挡图像具有更大的容忍度[34]。

经大量实验，我们发现 GICAH 和 GPCAH 算法在以下条件下识别性能较好：一种是状态个数 $N=5$，高斯概率混合成分个数 $M=3$；另一种是 $N=7$，$M=3$。表 4.7 显示了具体的数据。我们发现当观测向量维数 d 较低时，如 $d\in\{4,6,8\}$，GICAH 和 GPCAH 的识别率都很高，而且差别不大（见表 4.7 中 K3 的数据），这说明在低维情况下，两者都可以较好地对人脸进行建模。随着观测向量维数的增高（$d\geqslant 12$），两者的识别率都迅速下降，所以高维的观测向量不适合应用到人脸识别上。尽管如此，随着观测向量维数的增高，和 GPCAH 相比，GICAH 的识别率相对较高（见表 4.7 中 K8 的数据），这说明在对维数的适应性上，GICAH 要略好于 GPCAH。把表 4.7 中的数据绘成曲线得到图 4.21，从图中可以更方便地看出上述规律。

表 4.7　GICAH 和 GPCAH 的识别性能对比

方法	观测向量维数											均　值		
	4	6	8	12	16	20	24	28	32	36	40	K3	K8	K
ICA5	98.0	99.0	96.0	95.5	91.0	85.0	79.0	73.0	67.0	61.0	57.0	97.7	76.1	82.0
PCA5	97.5	99.0	97.0	91.5	86.5	85.5	78.5	72.5	64.0	58.5	47.0	97.8	73.0	79.8
ICA7	99.0	99.0	97.5	95.5	89.0	82.0	74.0	59.0	47.5	36.5	26.0	98.5	63.7	73.2
PCA7	98.0	96.5	95.5	89.5	81.5	77.0	64.0	52.0	38.5	29.5	16.5	96.7	56.1	67.1

注：① ICA5 表示用 Gabor+ICA+HMM 方法，状态个数为 5，4 种方法的高斯概率混合成分个数均为 3，其他类推。

② K3 表示对不同向量维数所对应的识别率的前 3 项（即 4～8）求均值，K8 表示对后 8 项（即 12～40）求均值，K 表示对所有的 11 项（即 4～40）求均值。

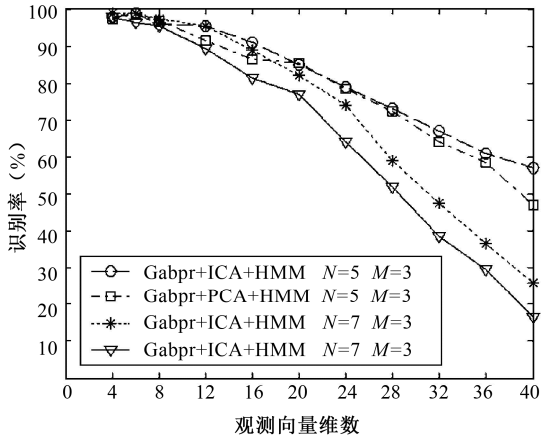


图 4.21　GICAH 与 GPCAH 的识别性能对比

4.5　本章小结

本章综述了人脸识别理论的概念和研究现状，讨论了其中的关键技术和难点以及应用和发展前景，并对人脸识别研究中应注意的问题提出了一些看法。由前面几节的介绍可以看到，人脸识别技术不断发展的过程，也就是对人脸的描述方式不断发展的过程。一方面，从最初的基于几何特征的人脸描述，到基于代数特征的整脸描述，到基于连接机制的弹性图描述，对人脸描述方式不断从低层次的图像特征向高层次的面部特征推进；另一方面，从代数特征把人脸仅仅作为普通图像来处理，到用 HMM 表示人脸面部特征的关联，再到用 3D 重建技术把人脸视为 3D 对象在 2D 平面上的投影，人脸形状、变化的先验知识被不断地加入到识别系统中。上述两个线索贯穿于人脸识别技术发展的始终，由于人脸识别任务的复杂性，使得任何单一方法都难以达到 100%的正确识别率。未来的系统在沿着上述两条线索发展的同时，还需要综合多种技术和方法形成组合系统。此外，由于每种生物特征鉴别系统各有优缺点，将不同种类的生物鉴别系统进行结合也越来越引起人们的重视，例如把人脸识别与指纹识别结合，或者与语音识别结合等。

虽然人类可以毫无困难地通过人脸而识别出某个人，但要建立一个能够完全自动进行人脸识别的人工智能系统却非常困难。困难的原因在于人类对视觉认知

机理的了解还很肤浅，还不知道如何用数学来准确描述认知现象。使用图像处理技术来进行人脸识别时，困难表现在：人脸光照模式的不确定性，人脸表情的多样性和人脸姿态的随意性。到目前为止，已经取得的研究成果离这一问题的彻底解决还有很大的距离。作者认为隐马尔可夫模型是描述复杂现象的有力工具，有希望帮助我们处理人脸识别问题。在本章中着重思考的问题是：寻找一组描述人脸特征的更合适的特征量，并把它作为隐马尔可夫模型的观测序列。这组特征量选取是否恰当，最终体现在识别算法是否能容忍光照变化、表情和姿态的变化，以及图像的部分遮挡等情况。同时，特征量的数目要适当，以保证识别的有效性和适度的计算价格。有人将 Gabor 小波用于对大脑皮层的视觉感知细胞的性态进行建模，表现了一定的合理性，它能较好地解释人的视觉对图像尺度的伸缩和方向变化的容忍度，所以可把它应用到人脸识别中。此外小波变换的多分辨率分析思想也近似符合人的视觉认知规律，因此也可以把它用于人脸表征。沿着上述思路，本章主要内容概述如下。

（1）提出了一种基于 Gabor 变换和隐马尔可夫模型的人脸识别方法。该算法先对人脸图像进行多分辨率的 Gabor 小波变换，然后在图像上放置一组网格结点，采用主元分析法对每个结点进行去相关、降维，最后形成 Gabor 脸。把 Gabor 脸的每个特征结作为观测向量，对隐马尔可夫模型进行训练，并把优化的模型参数用于人脸识别。分析了观测向量维数与识别率的关系，以及状态个数和高斯概率混合成分的个数对识别率的影响，测试了待识别图像经过部分遮挡后，算法的识别性能。对算法复杂度进行了分析，并同其他四种相关方法进行了比较。实验结果表明，本文方法识别率高，复杂度较低，对部分遮挡的图像具有较大的容忍度。

（2）提出了一种基于 Gabor 小波变换，独立元分析和隐马尔可夫模型的人脸识别方法。独立元分析法可以降低信号统计相关性，它的目标是寻找一个线性变换，把一系列随机变量表达成若干个统计独立的源信号的线性组合。该统计模型用高阶累积量来表示信号的互信息，并通过最小化互信息来确定所需的线性变换。由于独立元分析充分利用了信号的高阶统计量，在减少信号统计相关性的同时，降低了信号的维数，所以它可以更好地表达信号的本质特征。该方法仍先对人脸图像进行多分辨率的 Gabor 小波变换，采用独立元分析法对每个 Gabor 特征网格结点进行去相关、降维，最后形成特征结。然后把每个特征结作为观测向量，对隐马尔可夫模型进行训练，并把优化的模型参数用于人脸识别。实验结果表明，与其他相关方法相比，该方法识别率高，工程上易于应用。

本章参考文献

[1] Lades M, Vorbruggen J C, Buhmann J. Distortion invariant object recognition in the dynamic link architecture[J]. IEEE Transactions on computers, 1993, 42(3).

[2] Samaria F. Face recognition using hidden Markov model[D]. Cambridge：University of Cambridge, 1994.

[3] Nefian A. A hidden Markov model-based approach for face detection and recognition[D]. Georgia：Georgia Institute of Technology, 1999.

[4] Moghaddam B, Pentland A. Probabilistic visual learning for object representation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, 19(7).

[5] Moghaddam B, Pentland A. Beyond eigenfaces：probabilistic matching for face recognition[A]. In: Proceedings of the International Conference on Automatic Face and Gesture Recognition [C], Nara, 1998.

[6] Tenenbaum J, Silva V, Langford J. A global geometric framework for nonlinear dimensionality reduction[J]. Science, 2000, 290(5500).

[7] Roweis T, Saul K. Nonlinear dimensionality reduction by locally linear embedding[J]. Science, 2000, 290(5500).

[8] Helmuth L. Objection recognition：where the brain tells a face from a place[J]. Science, 2001, 292(5515).

[9] Rabiner L. A tutorial on hidden Markov models and selected application in speech recognition[J]. Proceedings of the IEEE, 1989, 77(2).

[10] Dempster A P, Rubin D B. Maximum likelihood from incomplete data via the EM algorithm[J]. Journal of Royal Statistical Society, 1977, 39(1).

[11] Rabiner L, Juang B. Fundamentals of Speech Recognition[M]. Englewood Cliffs：Prentice Hall, 1993.

[12] Field D. Relations between the statistics of natural images and the response properties of cortical cells[J]. Journal of the Optical Society of America A：Optics, Image Science, and Vision. 1987, 4(12).

[13] Daugman J. Complete discrete 2D Gabor transforms by neural networks for image analysis and compression[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1988, 36(7).

[14] Jones J, Palmer L. An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex[J]. Journal of neurophysiology, 1987, 58(12).

[15] Wiskott L, Fellous J, Kruger N. Face recognition by elastic bunch graph matching[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, 19(7).

[16] Phillips J, Moon H, Rizvi S. The FERET evaluation methodology for face-recognition algorithms[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(10).

[17] Duc B, Bigun J. Face authentication with Gabor information on deformable graphs[J]. IEEE Transactions on Image Processing, 1999, 8(4).

[18] Kruger V, Sommer G. Wavelet networks for face processing[J]. Journal of the Optical Society of American A：Optics, Image Science, and Vision, 2002, 19(6).

[19] Donato G, Bartlett M, Sejnowski T. Classifying facial actions[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1999, 21(10).

[20] Liu C J. Gabor-based kernel PCA with fractional power polynomial models for face recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26(5).

[21] Liu C J, Wechsler H. Gabor feature based classification using the enhanced Fisher linear discriminant model for face recognition[J]. IEEE Transactions on Image Processing, 2002, 11(4).

[22] Samaria F, Young S. HMM based architecture for face recognition[J]. Image and Computer Vision, 1994, 12(8).

[23] Othman H, Aboulnasr T. A separable low complexity 2D HMM with application to face recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(10).

[24] Helmuth L. Objection recognition：where the brain tells a face from a place[J]. Science, 2001, 292(5515).

[25] Duda R, Hart P, Stork D. Pattern classification, second edition[M]. New York： Wiley-Interscience, 2000.

[26] Juang B, Rabiner L. The segmental k-means algorithm for estimating the parameters of hidden Markov models[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1990, 38(9).

[27] William H, William T, Brian P. Numerical recipes in C：the art of scientific computing [M]. New York：Cambridge University Press, 1986.

[28] Haque M. A two-dimensional fast cosine transform[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1985, 33(6).

[29] Lee T. Image representation using 2D Gabor wavelets[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996, 18(10).

[30] Turk M, Pentland A. Eigenfaces for recognition[J]. Journal of Cognitive Neuroscience, 1991, 3(1).

[31] Karhunen J, Oja E. A class of neural networks for independent component analysis[J]. IEEE Transactions on Neural Networks, 1997, 8(3).

[32] Comon P. Independent component analysis, a new concept[J]. Signal Processing, 1994, 36(3).

[33] 邹谋炎. 反卷积和信号复原[M]. 北京：国防工业出版社, 2001.

[34] 曹林. 隐马尔可夫模型在人脸识别中的应用技术研究[D]. 北京：中国科学院电子学研究所, 2005.

第 5 章

人脸图像超分辨率重建

人脸超分辨率重建问题又称为虚幻脸（Face Hallucination）问题，是图像超分辨率重建技术中特殊的一类，人脸图像处理是数字图像处理中很典型的问题。在人脸超分辨率重建问题中，人脸的结构先验信息在重建策略中显得很重要，由于人脸具有高度规则的流形结构和高度相似性[1]，所以人脸图像的先验知识可以为人脸超分辨率重建技术提供丰富的指导信息。通常情况下，适用于一般图像的超分辨率重建算法不能很好地用于人脸超分辨率重建中，主要的原因在于一般图像不具有人脸的结构信息，不能利用人脸图像的一些固有特征信息去重建。目前用于人脸超分辨率重建的技术主要集中在基于学习的这类算法上，因为从人脸图像本身具有的固定特征来看，基于学习的超分辨率重建方法在这方面有着先天的优势，是研究的主流方向。基于学习的人脸超分辨率重建算法，即研究如何由输入的低分辨率人脸图像通过训练集重建出高分辨率人脸图像的过程[2]。

本章详细介绍了在人脸图像超分辨率重建中应用较广的基于学习的超分辨率算法的原理及优缺点，主要包括主元分析法、两步重建法等。并重点介绍所提出的基于分块 PCA 的单帧人脸图像超分辨率重建算法。对算法的思路和流程进行详细的阐述，该算法通过构建高低分辨率图像块训练集，学习训练集之间的映射关系，重建高分辨率人脸图像块，最后合成高分辨率人脸图像。

5.1　基于 PCA 的人脸超分辨率重建

基于 PCA（Principle Component Analysis，主元分析）的人脸超分辨率重建主要是构建高低分辨率人脸训练库，利用主元分析法来学习低分辨率人脸训练图像库和高分辨率人脸训练图像库之间的映射关系，利用映射关系得到重建的高分辨率人脸图像[3]。

5.1.1　PCA 算法原理

PCA（主元分析）是一种简单有效的方法，在降低数据的维数的同时，还能保留主要的数据信息，在模式识别领域被广泛使用。人脸具有结构上的相似性，一幅人脸可以通过一系列的样本人脸图像线性组合而成，因此可以利用 PCA 变换对人脸图像进行超分辨率重建[4]。

5.1.2　算法流程

基于 PCA 的人脸图像超分辨率重建算法的流程图如图 5.1 所示。

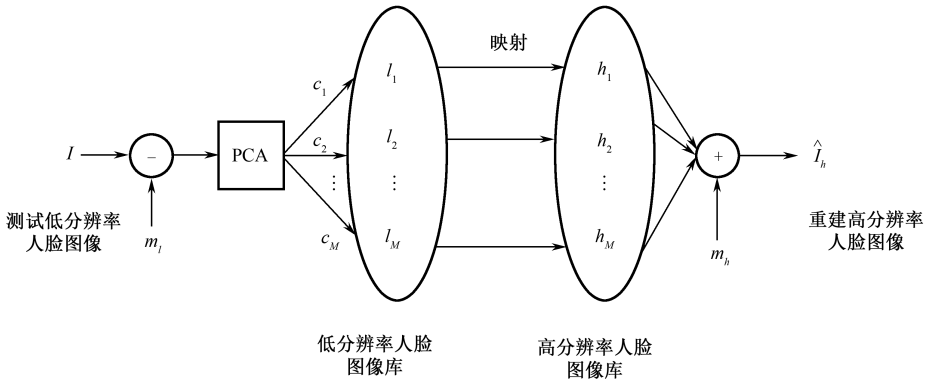


图 5.1　基于 PCA 的人脸超分辨率重建算法流程图

首先建立一个 $N\times M$ 的人脸训练库矩阵，$l=\{l_i\}_{i=1}^{M}$ 代表低分辨率图库，$h=\{h_i\}_{i=1}^{M}$ 代表高分辨率图库，N 是单幅图像的像素总和，M 是图像的个数（$N>>M$）。

计算平均脸的公式如下：

$$m_l=\frac{1}{M}\sum_{i=1}^{M}l_i \tag{5.1.1}$$

每张图片减去平均脸得到向量 L：

$$L=\left[l_1-m_l,\cdots,l_m-m_l\right] \tag{5.1.2}$$

计算协方差矩阵 C：

$$C=\sum_{i=1}^{M}(l_i-m_l)(l_i-m_l)^{\mathrm{T}}=LL^{\mathrm{T}} \tag{5.1.3}$$

因为协方差矩阵 C 的维数 $N\times N$ 是非常大的，通常情况下求 C 的特征值和特征向量是非常困难的，需要的内存大，计算速度会很慢。可以通过计算协方差矩阵的转置矩阵 $R=L^TL$ 的特征值 Λ 和特征向量 V 来解决此问题。

$$(L^TL)V=V\Lambda \tag{5.1.4}$$

其中特征向量 V 是 R 协方差转置矩阵的特征向量，特征值 Λ 是 R 的特征值。将上式两边同乘以 L，可得到：

$$L(L^TL)V=LV\Lambda \tag{5.1.5}$$

C 的正交特征向量计算如下：

$$E=LV\Lambda^{\frac{-1}{2}} \tag{5.1.6}$$

对于一张人脸图像 I 来说，可以将其投影到特征向量空间中，并用 K 个特征向量（特征脸）的加权线性组合来重建图像，特征脸如图 5.2 所示。

$$w=E^{\mathrm{T}}(I-m_l) \tag{5.1.7}$$

$$\hat{I}=Ew+m_l \tag{5.1.8}$$

图 5.2　特征脸

其中 w 是权值系数向量，$\hat{I}$ 是基于 PCA 重建的低分辨率人脸图像，特征向量的个数越多，重建效果越好，如图 5.3 所示。

图 5.3　特征脸个数对重建效果的影响

对上式可以进行等价变化。

$$\hat{I} = LV\Lambda^{\frac{-1}{2}}w + m_l = Lc + m_l = \sum_{i=1}^{M} c_i l_i + m_l \tag{5.1.9}$$

$$c = V\Lambda^{\frac{-1}{2}}w = [c_1, c_2, \cdots, c_M] \tag{5.1.10}$$

其中，c 代表了训练库人脸的权重系数，与测试人脸图像越相似性，则权重系数越大。而图像 $\hat{I}$ 则可以看成是训练库中所有样本的最优线性组合。因此，结合权重系数，并将低分辨率图像替换成一一对应的高分辨率训练人脸图像，得到最后重建的高分辨的人脸图像。

$$\hat{I}_H = \sum_{i=1}^{M} c_i h_i + m_h \tag{5.1.11}$$

其中 $h = \{h_i\}_{i=1}^{M}$ 是高分辨率人脸训练库，m_h 是高分辨率人脸图像的平均脸，而 $\hat{I}_H$ 是重建的高分辨率人脸图像。

基于 PCA 的人脸图像超分辨率重建算法的优点是原理简单，计算速度快，易于实现，但重建的高分辨率人脸图像的质量并不好，对人脸中心感兴趣的区域重建图像的效果较好，但是边缘较模糊，人脸图像细节损失较严重，图像轮廓周围出现”ghosting effect”。重建图像效果不好的原因，一是由于 PCA 算法本身具有的局限性，二是训练人脸图像库的数量和质量不是很好 [3]。

5.2　全局重建和残差补偿结合的人脸超分辨率重建

5.2.1　人脸超分辨率重建的约束条件

Liu[5]在他的文章中，首先提出了一个成功的人脸超分辨率算法需要服从的三个约束[6]。这些约束可以说是其全局模型与局部模型相结合的算法的根源。

（1）一致性约束：经过算法处理，得到的超分辨率结果在依据观测模型进行平滑和下采样之后必须非常地接近于原始输入。

（2）全局约束：人脸图像超分辨率重建的结果，即重建的高分辨率人脸图像必须拥有人脸的常见特征，如眼睛、嘴巴、鼻子、对称性等。

（3）局部约束：人脸图像超分辨率重建的结果，必须包含属于这张人脸图像的真实的局部细节特征。

这些约束看起来很平常且十分合理，但由于基于学习的超分辨率重建方法具有的不确定性，使得重建图像的结果不是很令人满意。第一个约束是比较容易满足的，通过对结果在处理中使用一个线性的限制就可以了。但相对而言，第二和第三个约束就显得比较难满足了。要同时满足三个约束要求，使得重建高分辨率人脸图像质量较高，这是非常重要的。因为如果没有对脸部细节特征的这些要求，可能得到的只是一个平滑的、接近均值脸的结果而已；如果没有对于全局的人脸这些限制，重建结果可能是充满噪声，甚至是错误的结果[3]。本章对文献[2]，马祥提出的全局重建和残差补偿法进行介绍，此方法也是在基于两步法框架的基础上提出的。

5.2.2 全局人脸重建

首先重建一幅全局人脸高分辨率图像，将二维人脸图像用一维向量来表示，设低分辨率人脸图像训练集为$[I_1, I_2, \cdots, I_n]$，对应的高分辨率人脸图像训练集为$[h_1, h_2, \cdots, h_n]$，其中 n 为训练集样本个数，I 为输入的低分辨率人脸图像。由于人脸结构的相似性，人脸图像可以由训练集图像线性组合进行逼近，即

$$I = \sum_{i=1}^{n} w_i I_i + e \tag{5.2.1}$$

式中，e 为逼近误差向量；w_i 为重建权值。这种线性组合关系是由人脸的结构相似性来决定的。重建权值向量 w 中的每个元素 w_i 体现了 I_i 对重建 I 所做的贡献，所有的权值 w_i 相加之和为 1，即

$$\sum_{i=1}^{n} w_i = 1 \tag{5.2.2}$$

使用欧式距离来度量重建误差，根据式（5.2.1），设 ε 为重建误差建立下式：

$$\varepsilon = \left\| I - \sum_{i=1}^{n} w_i I_i \right\|^2 \tag{5.2.3}$$

假设ε越小，重建就越成功，为求出w_i，结合式（5.2.2），建立一个成本函数：

$$w = \arg_{w_i} \min \varepsilon \tag{5.2.4}$$

这是一个在式（5.2.2）约束下的最小二乘问题。由于人脸结构的相似性，可以将I_i视为I的n个邻域向量，这样就将问题转化为求取I的n个邻域点权值的问题，使用 LLE 算法中求取重建权值的方法，有

$$\varepsilon = \left\| I - \sum_{i=1}^{n} w_i I_i \right\|^2 = \left\| \sum_{i=1}^{n} w_i (I - I_i) \right\|^2 = \sum_{i=1}^{n} \sum_{k=1}^{n} w_i w_k C_{ik} \tag{5.2.5}$$

构造一个局部协方差矩阵C，C中元素$C_{ik} = (I - I_i)^{\mathrm{T}} (I - I_k)$，使用托格朗日乘法，在保证所有的$w_i$相加和为1的约束下可得到

$$w_i = \sum_{k=1}^{n} C_{ik}^{-1} / \sum_{l=1}^{n} \sum_{m=1}^{n} C_{lm}^{-1} \tag{5.2.6}$$

式中，l、m均为正整数；C_{ik}^{-1}表示逆矩阵C^{-1}中第i行第k列元素。在实际运算中，C可能是一个奇异矩阵，此时必须正则化 C 在实际的算法中常采用一个更快捷的方法，即求线性系统方程$Cw = 1$，最后令解得的权值满足相加和为 1。

利用权值，可以得到输入图像I的重建图像

$$\tilde{I} = \sum_{i=1}^{n} w_i I_i + e \tag{5.2.7}$$

由式（5.2.3）可知，重建的效果依赖于

$$\varepsilon = \left\| I - \tilde{I} \right\|^2 \tag{5.2.8}$$

将式（5.2.7）中低分辨率训练集图像I_i，置换为高分辨率训练集图像h_i，结果为

$$H = \sum_{i=1}^{n} w_i h_i \tag{5.2.9}$$

H即为初步的全局重建结果，此时重建的图像质量很差，细节信息不足，图像较模糊。因此需要采用残差补偿的方法进行细节信息补偿。

5.2.3　残差补偿

残差补偿的步骤如下。

首先计算残差补偿的输入图像，即用低分辨率图像减去全局重建的高分辨率

人脸图像的下采样 $S = I - D(H)$。并且生成高分辨率残差训练人脸图像库 $H_m = h_m - H$ 和低分辨率残差训练人脸图像库 $L_m = I_m - D(H)$。

然后将图像分成具有重叠像素的固定大小的图像块 $\{L_m(i,j)\}_p$ 和 $\{H_m(i,j)\}_p$，对于每个残差图像块，利用 PCA 的方法求权重系数，方法在上一小节中讲过，合成高分辨率残差重建图像块。

接下来将高分辨率残差图像块按照在人脸中的位置，拼接成人脸残差图像 $\tilde{R}$。

最后将全局重建的人脸图像 H 和残差图像 $\tilde{R}$ 相加，即高分辨率重建人脸图像。

如果得到的高分辨率图像效果仍然不好，需要对上述过程进行不断的迭代，直到取得较满意的效果。这种方法可以对图像的细节信息进行不断的修改，比基于 PCA 重建的人脸图像的效果更好，人脸细节变得清晰。

总之，基于全局和局部的两步法重建算法，重建图像的效果较好，但是计算量很大，需要不断的迭代，直到重建图像效果较好为止。

5.3 基于分块 PCA 的单帧人脸图像超分辨率重建

根据上面两节提出的两种算法的优缺点，提出本节的算法，即基于分块 PCA 的人脸图像超分辨率重建算法，是针对单帧人脸图像的重建。此算法的思想是，首先构建高低分辨率人脸图像库，然后将测试人脸图像和高低分辨率人脸图像分块，利用 PCA（主元分析法）学习训练人脸图像块库之间的映射关系，利用此关系来指导得到重建的高分辨率图像块，最后合成得到一幅高分辨率人脸图像。

5.3.1 图像分块策略

假定一幅高分辨率人脸图像 I_{H} 的像素为 $M_1 \times N_1$，一幅低分辨率图像 I_{L} 的像素为 $m_1 \times n_1$，下采样因子是 p，则 $m_1=M_1/p$，$n_1=N_1/p$。

对 I_{H} 和 I_{L} 采用固定大小的有重叠像素的分块方式，上下左右相邻图像块之间有重叠的像素。为了说明一般的情况，我们假设图像分块的长和宽是不相等的，高分辨率图像中每个块的大小为 $M \times N$，相邻块之间水平重叠像素为 T 和垂直方向重叠像素为 $\overline{T}$。低分辨率图像中每个块的大小为 $m \times n$，相邻块之间水平方向重叠像素为 t 和垂直方向重叠像素为 $\overline{t}$（$t=T/p$，$\overline{t}=\overline{T}/p$）。最后得到的高分辨率

图像块和低分辨率图像块分别用 I_{H}^{P} 和 I_{L}^{P} 表示，如图 5.4 所示。

一个高分辨率图像分块可表示为：

$$I_{\mathrm{H}}^{P}[i,j], i=1,2,\cdots,I;\ j=1,2,\cdots,J \tag{5.3.1}$$

一个低分辨率图像分块可表示为：

$$I_{\mathrm{L}}^{P}[i,j], i=1,2,\cdots,I,\ j=1,2,\cdots,J \tag{5.3.2}$$

其中 i 和 j 分别表示图像块在整幅图像中的横坐标和纵坐标。I 和 J 是一幅图像在水平和垂直方向分完块的个数，计算公式分别如下：

$$I=1+\frac{M_1-M}{M-T}=1+\frac{m_1-m}{m-t} \tag{5.3.3}$$

$$J=1+\frac{M_1-M}{M-\overline{T}}=1+\frac{m_1-m}{m-\overline{t}} \tag{5.3.4}$$

其中 $I_{\mathrm{H}}^{P}[i,j]$ 表示高分辨率图像 I_{H} 中第 i 行第 j 列的图像块。

一幅图像经过重叠分块后总的分块个数等于 I 和 J 相乘。

图 5.4、图 5.5 表示出了高分辨率人脸图像和低分辨率人脸图像分块之间的关系。

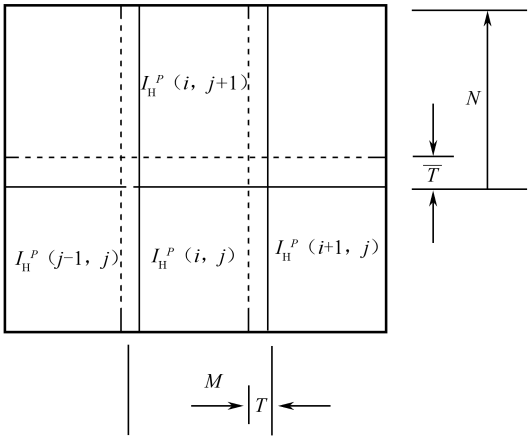


图 5.4　图像分块

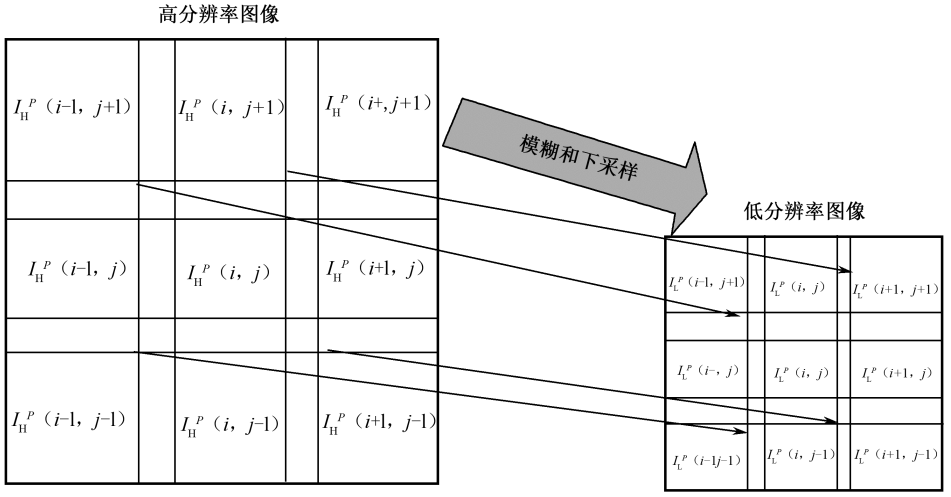


图 5.5　高低分辨率图像分块之间的关系

5.3.2　训练库生成策略

训练库包括很多对高低分辨率人脸图像，训练库人脸图像的质量，种类和训练库的大小对超分辨率重建结果有很大的影响，如果选择恰当，最后重建图像的质量就越好[1,7]。

图像超分辨率重建技术的退化模型，对自然界的实际事物，人们是不可能用理想的成像设备获得高分辨率图像的，是因为现实中由于成像系统的内在条件（系统本身固有的物理属性、感光元件的采样率）的限制和外部环境因素（如运动模糊、噪声等）的干扰，总是不可避免的引入一些几何形变、模糊、噪声等干扰因素，加上有时摄像头离拍摄物拍摄距离较远，实际得到的图像是低分辨率的。

通过一幅高分辨率图像得到一幅低分辨率图像的降质模型为：

$$Y = D \times H \times M \times X + n \tag{5.3.5}$$

其中 Y 表示一幅高分辨率图像，X 表示一幅低分辨率图像，D 表示下采样，H 表示模糊，M 表示几何形变，n 表示噪声。

在本章方法中将高分辨率人脸训练图像通过下采样、高斯模糊得到一一对应的低分辨率人脸训练图像，这里不考虑形变和噪声的问题。得到的高低分辨率人脸训练图库如图 5.6 和图 5.7 所示。

图 5.6　高分辨率人脸图像训练库

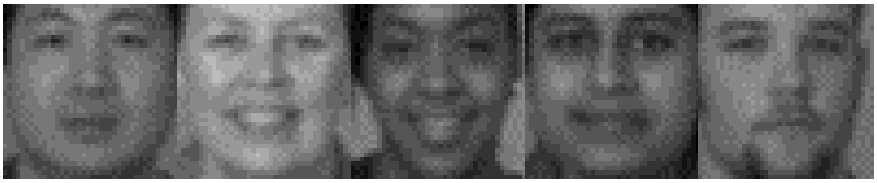


图 5.7　对应的低分辨率人脸图像训练库

5.3.3　算法流程

本算法是基于分块的思想，如图 5.8 所示，即为待重建的低分辨率人脸图像的每个图像块，在低分辨率训练集中求得映射权重系数，利用这些样本在高分辨率训练集中对应的高分辨率样本来重建当前图像块，最后完成整个人脸图像的重建。本算法的流程图如图 5.9 所示。

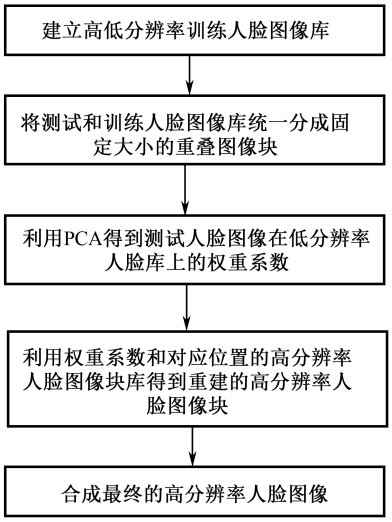


图 5.8　分块 PCA 算法思想

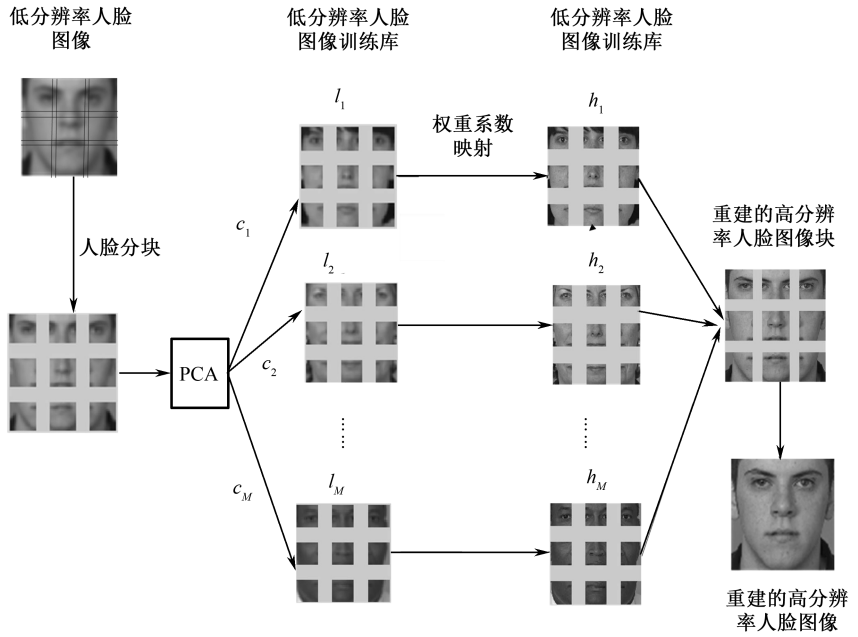


图 5.9　基于分块 PCA 的人脸图像超分辨率重建算法流程

基于分块 PCA 的人脸图像超分辨率重建算法主要包括下面几个步骤。

1．构建人脸样本库

5.3.2 节中具体介绍过如何构建训练样本库，这里就不再介绍了。最后得到低分辨率训练人脸图像库 $l=\{l_i\}_{i=1}^{M}$ 和高分辨率训练人脸库 $h=\{h_i\}_{i=1}^{M}$，M 是训练库样本的个数。

2．图像分块

按照 5.3.1 节介绍的图像分块策略，将测试低分辨率人脸图像和高低分辨率训练图像分成固定大小的重叠的人脸图像块，得到高低分辨率训练人脸图像块库分别表示成 $\{l_{ij}\}_{j=1}^{k}$ 和 $\{h_{ij}\}_{j=1}^{k}$，k 表示一幅图像中图像块的个数。

3．PCA 求权重系数

由于人脸图像具有结构上的相似性，因此一幅人脸图像可以通过一系列的样本人脸图像线性组合而成[4]，故基于 PCA 的人脸超分辨率重建的步骤如下所示：

首先求第 j 个低分辨率训练图像块库的平均值 m_j

$$\mathrm{m}_j = \frac{1}{M}\sum_{i=1}^{M} I_{ij} \tag{5.3.6}$$

每个图像块减去平均值：

$$L_j = [l_{1j} - m_j, \cdots, l_{mj} - m_j] = [l'_{1j}, \cdots, l'_{mj}] \tag{5.3.7}$$

然后求协方差矩阵：

$$C_j = \sum_{i=1}^{M} (l_{ij} - m_j)(l_{ij} - m_j)^{\mathrm{T}} = L_j L_j^{\mathrm{T}} \tag{5.3.8}$$

计算协方差矩阵 C_j 的特征值 Λ_j 和特征向量 V_j。

根据公式

$$V_j V_j^T = I_j, j = 1,2,\cdots,k \tag{5.3.9}$$

$$R_j V_j = V_j \Lambda_j \tag{5.3.10}$$

其中 I_j 单位矩阵，将特征值进行降序排序，Λ_j 是前面较大的特征值，特征值 Λ_j 对应的特征向量 V_j。较大的特征值对应的特征脸表示的是图像的轮廓信息，特征值小的特征值代表图像的细节信息。

接下来求在该低分辨空间下的特征向量矩阵：

$$E_j = L_j V_j \Lambda_j^{\frac{-1}{2}} \tag{5.3.11}$$

对于测试人脸图像的第 j 个图像块 x_j，将其投影到特征向量矩阵上，可以得到投影系数：

$$w_j = E_j^{\mathrm{T}}(x_j - m_j) \tag{5.3.12}$$

第 j 个测试低分辨率人脸图像块经 PCA 重建得到的人脸图像块：

$$r_j = E_j w_j + m_j = L_j V_j \Lambda_j^{\frac{-1}{2}} w_j + m_j = L_j S_j + m_j \tag{5.3.13}$$

其中

$$S_j = V_j \Lambda_j^{\frac{-1}{2}} w_j = [S_{j1}, \cdots, S_{jM}] \tag{5.3.14}$$

式（5.3.14）可以重新定义为

$$r_j = L_j S_j + m_j = \sum_{i=1}^{M} S_{ji} l'_{ij} + m_j \tag{5.3.15}$$

高分辨率人脸图像重建块可以由 M 个训练人脸图像块库进行加权得到。S_j 是权重系数，如果训练人脸图像和测试人脸图像越相似，权重系数就越大，相反，如果越不相似，权重系数就越小，如图 5.10 所示。

图 5.10　人脸图像可以由训练人脸图像库加权线性合成

4．重建高分辨率人脸图像

由 PCA 重构特性，可以重构出对应的高辨率人脸图像块。用相同位置的高分辨率训练图像块 h_{ij} 代替低分辨率训练图像块 l_{ij}，和用高分辨率训练人脸图像块库的平均值 M_j 代替低分辨率训练人脸图像块库的平均值 m_j。

结果是

$$R_j = \sum_{i=1}^{M} S_{ji} h_{ij}^{'} + M_j \tag{5.3.16}$$

式中，R_j 是重建的第 j 个高分辨率人脸图像块。

对测试人脸的所有图像块重复进行上述过程的计算，直到求出所有位置的高分辨率图像块为止。

将重建的高分辨率图像块按照原来在人脸中的位置合成对应的高分辨率重建图像，重叠过程中如果遇到重叠像素，则取各个高分辨率图像块在该像素点的灰度值的均值来作为重叠像素最终的灰度值。

5.4　本章小结

本章介绍了基于学习的人脸超分辨率方法的两种代表性的算法，两种算法各有优缺点。PCA 重建算法对人脸中心感兴趣区域超分辨率重建的效果较好，但是其他部位尤其脸部边缘很模糊，脸部细节损失得较严重；全局和局部相结合的方法重建人脸图像效果较好，但是计算速度慢，需要不断的迭代。基于这两种算法的优缺点，接着提出了一种基于分块 PCA 的单帧人脸超分辨率算法。此算法用人脸图像块位置信息表征人脸图像的全局结构，图像块内容表征人脸图像的细节。利用 PCA（主元分析法）提取训练人脸库的特征，构建权重系数，重建高

分辨率图像。这种方法继承了基于学习的人脸超分辨率重建方法的优点，克服了邻域嵌入法对邻近值 K 的依赖，比两步法重建算法计算速度更快。

本章参考文献

[1] 刘良辰. 基于整体到局部的分离式人脸超分辨率重建方法研究[D]. 重庆：重庆大学，2010.

[2] 马祥, 齐春. 全局重建和位置块残差补偿的人脸图像超分辨率算法[J]. 西安交通大学学报, 2010, 44（4）: 9-12.

[3] 沈华. 基于插值和主元素分析的人脸超分辨率算法研究[D]. 湖南:湖南大学,2010.

[4] 马祥, 刘军辉. 基于 PCA 与残差补偿的人脸超分辨率算法[J]. 计算机工程, 2012, 38（13）: 196-198.

[5] C Liu, H Y Shun, C H Zhang. A two-step approach to hallucinating faces: Global parametric model and local nonparametric model[J]. IEEE Computer Society Conference on Computer Vision and Pattern Recognition,2001, 1: 192-198.

[6] 卓力, 王素玉, 李晓光.图像/视频的超分辨率复原[M]. 北京：人民邮电出版社, 2011, 1.

[7] 邱一雯. 基于学习的视频超分辨率重建算法研究与实现[D]. 南京:南京邮电大学,2012.

第 6 章

Kinect 人体动作识别

人体动作识别在虚拟现实、人机交互领域也是一个重要的课题。随着虚拟现实（Virtual Reality，VR）时代的来临，自然人机交互（Natural User Interface，NUI）应运而生，而正是在这种虚拟与现实世界重叠的背景下，微软公司的 Kinect 深度传感器大放异彩。虚拟现实[1]（又译作灵境、幻真）是随着计算机性能发展到一定程度产生的，通过计算机以及相关学科技术模拟出一个数字的三维虚拟世界，用户采用一些特定的人机交互设备与虚拟环境中的对象进行交互，从而通过视觉、触觉和听觉等产生置身于相应的真实环境中的虚幻感、沉浸感以及真实感。在虚拟现实的环境中进行人体行为的识别，是计算机视觉领域研究的一个热点问题。

人体行为的复杂性不言而喻，相比于其他模式识别的研究更具挑战性。因此，行为的表述即特征提取显得尤为重要，以往研究大多关注边缘、形状、姿态、轨迹等静态特征以及光流[2]、运动能量等动态特征，但是在比较复杂场景及遮挡、噪声等干扰下鲁棒性面临挑战。

本章将介绍通过引入 Kinect 深度传感器提取人体动作的骨骼特征，并结合机器学习的方法进行样本训练以及动作识别。同时，也介绍了人体行为的三维时空直方图特征以及二维轮廓特征的行为描述方法。本章在二维视频数据中引入了空间的概念，即第三维是时间数据，然后探索时空中梯度方向以及空间角度来描述人体行为。

6.1　基于 Kinect 骨骼空间几何角度的动作识别

本小节提出了一种基于深度传感器提取人体骨骼空间角度信息的动作识别方法，研究了人体骨骼结构、骨骼关节点位置信息以及人体动作所具有的骨骼空间几何角度等特征。

与传统的彩色摄像头不同，Kinect 深度传感器能够提供第三维深度数据。它能克服彩色摄像头易受光线等外界干扰的缺点，准确追踪到视野范围内的人体[3]。当人做出不同动作时，相应的关节与骨骼具有不同的位置和角度信息。因此，追踪关节点的坐标和探索骨骼的空间几何角度信息将提供非常可靠和直接的方法来实现人体动作识别。

6.1.1　人体骨骼信息获取

Kinect 深度传感器能够同时提供彩色图像数据、深度图像数据和骨骼数据，并且能够检测出人体 20 个骨骼关节点[4]。因此我们可以获取到人体骨骼数据，然后形成骨骼拓扑结构。

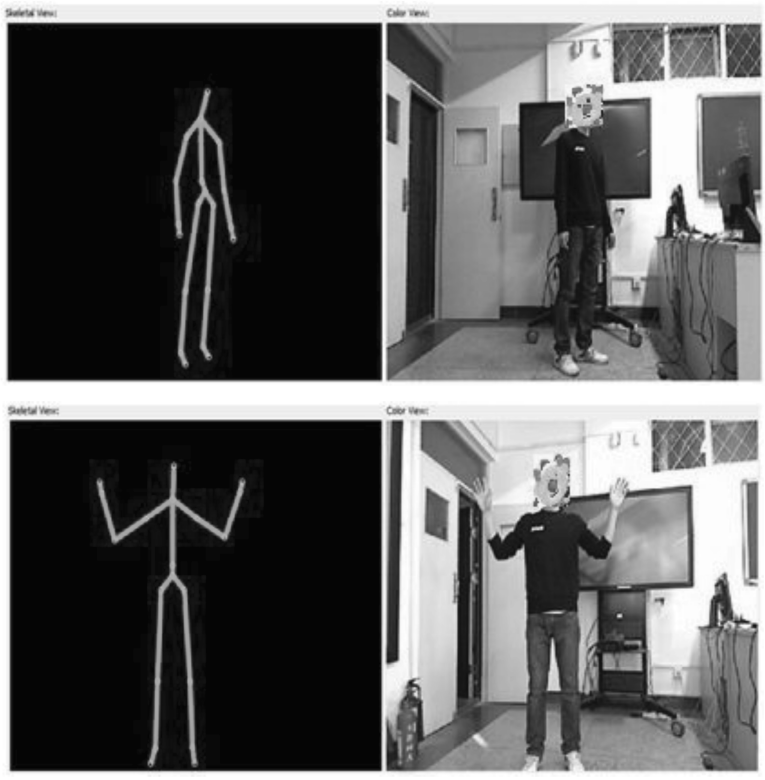
开启 Kinect 骨骼数据流后，当有人出现在 Kinect 有效视野范围内时，Kinect 将迅速检测出人体并实时跟踪人体骨骼。其步骤如下：

（1）Kinect 获取人体 20 个关节点相对于 Kinect 骨骼坐标系的三维坐标数据；

（2）根据坐标变换将骨骼三维坐标系变换到屏幕二维坐标系；

（3）根据人体骨骼拓扑结构构建人体骨架，并使其可视化。

图 6.1（a）为 Kinect 主动追踪的视野范围内的人体骨骼图，图 6.1（b）为场景的彩色视频图像。



（a）人体骨骼拓扑结构　　（b）彩色视频图像

图 6.1　Kinect 主动追踪人体骨骼图

6.1.2　骨骼空间角度特征提取

1．提取感兴趣的关节点

通过对人体运动时骨骼旋转特性以及人体骨骼拓扑结构的研究，我们将 Kinect 传感器检测出的人体 20 个关节点进行分层：

第一层：身体躯干关节点。支撑起整个人体的第一层骨骼关节点构成人体的躯干，该层主要包括头、左右肩、脊椎等 8 个关节点（如图 6.2 中用数字 1 标记的关节点）。人体某些动作所需的特征信息部分来自于这层关节点。

第二层：四肢关节点。人体的大多数动作（如挥手、出拳、踢腿等）主要都是靠人的四肢来表达的，它们包含了大量人体运动和姿势的特征信息。因此我们

将四肢分为第二层，包括左右肘、腕，左右膝、踝 8 个关节点（如图 6.2 中用数字 2 标记的关节点）。

第三层：手与脚。剩余的左右手、脚这 4 个关节点，我们将其归为第三层（如图 6.2 中用数字 3 标记的关节点）。手掌和脚在我们所研究的人体动作中所起的作用非常微小，对识别所提供的特征信息可以忽略。

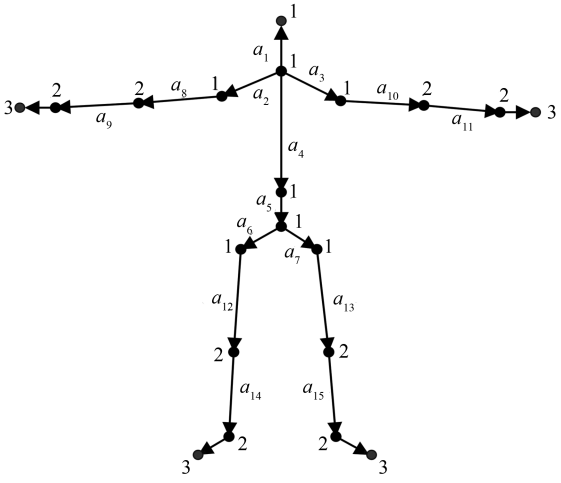


图 6.2　分层关节点与定义骨骼向量

为了降低特征数据维数和提高计算效率，因此进一步提取识别动作所感兴趣的关节点，对于信息量很少的关节点予以去除。通过以上的分层分析，我们只提取第一层和第二层这 16 个重要的关节点进行下一步工作。

2．定义骨骼向量

从前面所提取得到的 16 个关节点中，根据人体结构学原理，将每两个关节点连接组成一段骨骼，并将其定义为一个向量。将第一层关节点连接组成人体躯干，共有 7 个骨骼向量，分别将这 7 个向量标记为$\{\boldsymbol{a_1}, \boldsymbol{a_2}, \cdots, \boldsymbol{a_7}\}$；第二层关节点连接组成四肢，共有 8 个骨骼向量，分别标记为$\{\boldsymbol{a_8}, \boldsymbol{a_9}, \cdots, \boldsymbol{a_{15}}\}$。如图 6.2 所示，15 个骨骼向量分别对应于图中所给出的标记。

3．骨骼向量方向余弦特征提取

当人做出不同的动作时，对于人体的每一段骨骼而言，都具有不同的位置和角度信息。因此，可以利用定义的 15 个骨骼向量的方向余弦特征来表征某一类动作。

现取出右手的肩关节到肘关节这个骨骼向量（即标号为 a_8 的向量）进行详细算法分析：

设 Kinect 获取肩关节的三维坐标为(x_1,y_1,z_1)，肘关节的三维坐标为(x_2,y_2,z_2)，如图 6.3 所示。由此，该骨骼向量可表示为

$$a_8=\{x_2-x_1,y_2-y_1,z_2-z_1\} \tag{6.1.1}$$

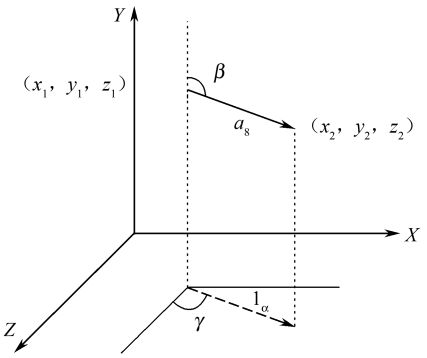


图 6.3　骨骼向量的方向余弦计算

设向量 a_8 与 Kinect 坐标系的三个方向角分别为α、β、γ，如图 6.3 所示。由此，我们可以得到该骨骼向量的三个方向余弦值，公式如下：

$$\cos\alpha=\frac{x_2-x_1}{\sqrt{(x_2-x_1)+(y_2-y_1)+(z_2-z_1)^2}} \tag{6.1.2}$$

$$\cos\beta=\frac{y_2-y_1}{\sqrt{(x_2-x_1)+(y_2-y_1)+(z_2-z_1)^2}} \tag{6.1.3}$$

$$\cos\gamma=\frac{z_2-z_1}{\sqrt{(x_2-x_1)+(y_2-y_1)+(z_2-z_1)^2}} \tag{6.1.4}$$

其中

$$\sqrt{(x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2}\neq 0 \tag{6.1.5}$$

通过上述算法，依次将已定义的 15 个骨骼向量的方向余弦值作为特征进行提取。

4．骨骼夹角余弦特征提取

另一方面，对于人体不同的动作而言，通过某一个关节点连接的两段骨骼之间有着不同的空间夹角关系。因此，还可以通过提取定义的 15 个骨骼向量之间

的夹角余弦特征来描述人体行为。如图 6.4 所示，图中两实线所成的夹角θ_1、θ_2、θ_3即为两骨骼之间的空间夹角。

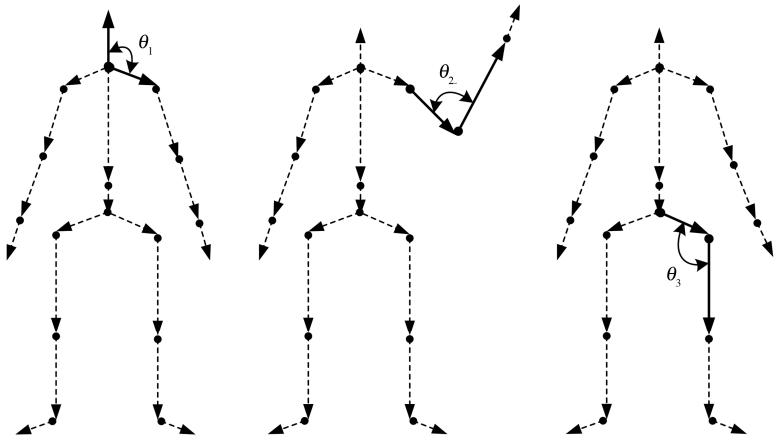


图 6.4　两骨骼之间的夹角

以某两段骨骼向量为例计算它们空间夹角的余弦值。在空间中，设骨骼向量$\boldsymbol{a_1}$用坐标表示为(l_1,m_1,n_1)，骨骼向量$\boldsymbol{a_2}$用坐标表示为(l_2,m_2,n_2)，如图 6.5 所示，于是可得到空间中这两骨骼的夹角余弦，公式如下：

$$\cos\theta=\frac{l_1l_2+m_1m_2+n_1n_2}{\sqrt{l_1^2+m_1^2+n_1^2}\sqrt{l_2^2+m_2^2+n_2^2}} \tag{6.1.6}$$

其中

$$\sqrt{l_1^2+m_1^2+n_1^2}\neq 0 \text{ 且 } \sqrt{l_2^2+m_2^2+n_2^2}\neq 0 \tag{6.1.7}$$

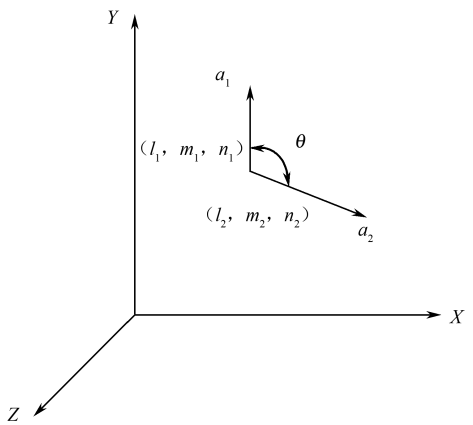


图 6.5　骨骼之间夹角余弦值计算

根据上述算法，依次计算每两段骨骼之间的夹角余弦值作为特征进行提取。

5．Kinect 俯仰角的调节

单个 Kinect 设备不能对人体进行全方位的观测，人体站立的位置与 Kinect 设备镜头平面构成的角度的改变也会影响到关节点检测的结果。为了得到合适阈值，需要确定 Kinect 仰角。假设人的位置与镜头平面构成的夹角为θ，人所在位置的平面为α，镜头平面为β，两平面的法向量分别为$\vec{m}=(x_1,y_1,z_1)$，$\vec{n}=(x_2,y_2,z_2)$。

（1）法向量反向时

当两平面的法向量（图 6.6）反向时，二面角$\alpha-l-\beta$的值为

$$\theta=<\vec{m},\vec{n}>=\arccos\frac{\vec{m}\cdot\vec{n}}{|\vec{m}|\cdot|\vec{n}|} \tag{6.1.8}$$

（2）法向量同向时

当两平面的法向量（图 6.7）同向时，二面角$\alpha-l-\beta$的值为

$$\theta=<\vec{m},\vec{n}>=\pi-\arccos\frac{\vec{m}\cdot\vec{n}}{|\vec{m}|\cdot|\vec{n}|} \tag{6.1.9}$$

其中，$\vec{m}\cdot\vec{n}=x_1x_2+y_1y_2+z_1z_2$，$|\vec{m}|=\sqrt{x_1^2+y_1^2+z_1^2}$，$|\vec{n}|=\sqrt{x_2^2+y_2^2+z_2^2}$。

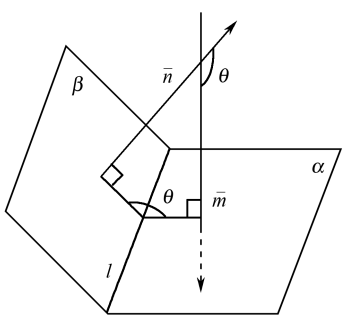


图 6.6　法向量反向时

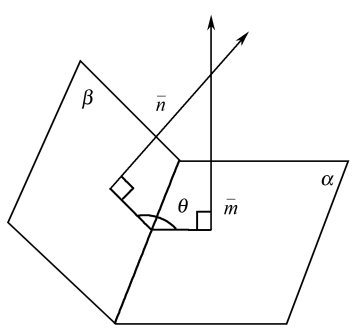


图 6.7　法向量同向时

对于其中空间平面的法向量可以使用向量外积法求解，如图 6.8 所示。

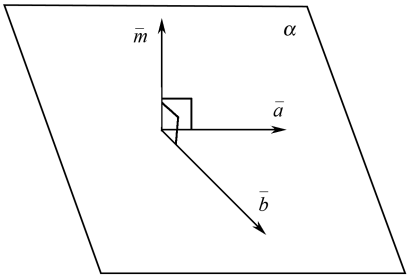


图 6.8　求解平面法向量

平面 α 的法向量为 $\vec{m}$，假设向量 $\vec{a}=(x_1,y_1,z_1)$ 和 $\vec{b}=(x_2,y_2,z_2)$ 时平面上不平行的任意非零向量，则

$$\vec{m}=\vec{a}\times\vec{b} \tag{6.1.10}$$

其中

$$\vec{a}\times\vec{b}=\begin{vmatrix} i & j & k \\ x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \end{vmatrix}=\left(\begin{vmatrix} y_1 & z_1 \\ y_2 & z_2 \end{vmatrix},-\begin{vmatrix} x_1 & z_1 \\ x_2 & z_2 \end{vmatrix},\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix}\right) \tag{6.1.11}$$

其中，法向量的方向可由右手准则确定。

根据上面的理论基础，观测映射到图像中的关节点的变化可知，Kinect 镜头观测平面夹角为 0～60° 时可以取得较好的观测效果。

6.1.3　多分类支持向量机

1．支持向量机

支持向量机[5]（Support Vector Machines，SVM）由 Vapnik 首先提出，是一种两类分类模型。它的主要思想是建立一个分类超平面作为决策曲面，使得两类之间的隔离边缘被最大化。它基于统计学习的理论，确切地说，支持向量机是结构风险最小化的近似实现。

常见的二分类支持向量机模型如下：

已知训练集

$$T=\{(x_1,y_1),\cdots,(x_l,y_l)\}\in(X\times Y)^l \tag{6.1.12}$$

其中，$x_i\in X=R^n$，$y_i\in Y=\{1,-1\}(i=1,2,\cdots,l)$，$x_i$ 为特征向量。

（1）对于线性分类问题，即当训练集线性可分时，此时构造并求解约束最优

化问题：

$$\min_{w,b}\frac{1}{2}\|w\|^2 \tag{6.1.13}$$

$$\text{s.t. } y_i(w\cdot x_i+b)-1\geqslant 0, i=1,2,\cdots,l \tag{6.1.14}$$

求得最优解 w^*,b^*

由此得到分离超平面为

$$w^*\cdot x+b^*=0 \tag{6.1.15}$$

分类决策函数为：

$$f(x)=\text{sign}(w^*\cdot x+b^*) \tag{6.1.16}$$

（2）对于非线性分类问题，即当训练集线性不可分时，我们的做法是首先定义一个核函数 $K(x_i,x_j)$ 和适当的参数 C，然后构造并求解最优化问题:

$$\min_{\alpha}\frac{1}{2}\sum_{i=1}^{j}\sum_{j=1}^{l}y_iy_j\alpha_i\alpha_jK(x_i,x_j)-\sum_{j=1}^{l}\alpha_j \tag{6.1.17}$$

$$\text{s.t. } \sum_{i=1}^{l}y_i\alpha_i=0 \tag{6.1.18}$$

其中，$0\leqslant\alpha_i\leqslant C, i=1,2,\cdots,l$，由此，得到最优解：$\alpha^*=(\alpha_1^*,\cdots,\alpha_l^*)^{\mathrm{T}}$。

选取α^*的一个正分量 $0<\alpha_j^*<C$，并据此计算阈值

$$b^*=y_j-\sum_{i=1}^{l}y_i\alpha_i^*K(x_i-x_j) \tag{6.1.19}$$

因此分类决策函数为

$$f(x)=\text{sign}\left[\sum_{i=1}^{l}\alpha_i^*y_iK(x,x_i)+b^*\right] \tag{6.1.20}$$

2．常用核函数

对于不同的研究内容及训练集而言，采用不同的核函数进行训练，会有不同的识别分类效果。常见的几种核函数如下。

（1）多项式核函数：

$$K_{\text{poly}}(x,x_i)=(\gamma x^{\mathrm{T}}x_i+r)^p \tag{6.1.21}$$

（2）径向基核函数（RBF 核函数）：

$$K_{\text{rbf}}(x,x_i)=\exp(-\gamma\|x-x_i\|^2) \tag{6.1.22}$$

（3）两层感知机核函数（Sigmoid 核函数）：

$$K_{\text{sigmoid}}(x,x_i)=\tanh(\gamma x^{\mathrm{T}}x_i+r) \tag{6.1.23}$$

3．多分类支持向量机

SVM 算法最初是为解决两类分类问题，而本章的目标是需要处理多类问题，因此我们必须采用多分类支持向量机（Multi-class SVM）算法。主要思想是通过组合多个二分类器来构造多分类器的模型，常用的构造方法有一对多、一对一、层次支持向量机和有向无环图等。

一对一（one-versus-one）算法原理：对于有 k 个类别的分类问题，依次选出任意两类样本组成一个二分类问题，并设计一个 SVM 模型。因此，该 k 个类别的分类问题总共需要设计 $k(k-1)/2$ 个 SVM 模型。当输入一个未知样本进行预测分类时，由之前设计的每个分类器对它进行预测，最后得票最多的类别即为该未知样本的类别。

6.1.4　训练与识别结果分析

1．数据采集

定义五种不同的人体动作进行研究：行走、挥手、拍手、弯腰、踢腿，如图 6.9 所示。

（a）行走　　（b）挥手　　（c）拍手

（d）弯腰　　（e）踢腿

图 6.9　五种不同人体动作示例

为了使训练数据有效，分别采集了多个人的动作骨骼数据，并区分男、女及高、矮不同差异的人。每个人每种动作分别做三次实验：正面距 Kinect 1.5m、侧身距 Kinect 1.5m、正面距 Kinect 3m。采集到的样本部分用于模型训练，剩余样本用于识别测试。由此，建立了本章的训练数据集和测试集，如表 6.1 所示。例如，行走动作，训练集总共有 1400 帧骨骼数据，每一帧骨骼数据包含以下内容：一个动作的 15 个骨骼向量，每个骨骼向量有三个方向余弦值；以及 18 个骨骼之间的夹角余弦值。

表 6.1　训练数据集和测试集

人体动作	行走	挥手	拍手	弯腰	踢腿
训练集/帧	1400	1320	1270	1250	1240
测试集/帧	900	1000	900	800	850

2．SVM 参数寻优

取每种动作 200 帧数据，则五种动作总共 1000 帧数据作为测试集，首先采用遗传算法（GA）对该测试集进行 SVM 最佳参数寻优，图 6.10 是惩罚参数 *c*（cost）与核函数参数 *g*（gamma）的参数寻优结果。

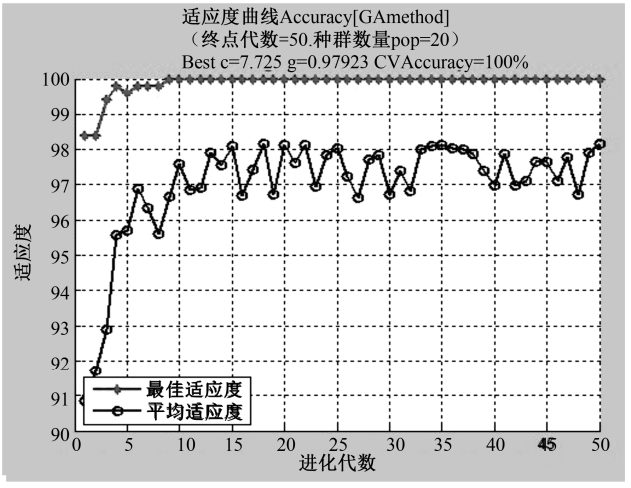


图 6.10　SVM 参数寻优的适应度（准确率）曲线

从图 6.10 中得到最优参数结果是：*c*=7.73；*g*=0.98。

在 6.1.3 节中已经介绍了不同核函数的几种 SVM 算法，接下来将对这 1000 帧测试集分别利用这几种核函数和得到的最优参数进行模型训练和测试集预测

分类。表 6.2 中给出了它们分类准确率的对比。

表 6.2　SVM 不同核函数性能对比

采用不同核函数	多项式核函数	RBF 核函数	Sigmoid 核函数
分类准确率（%）	95.87	89.48	27.03

由表 6.2 中的数据看出，多项式核函数得到的测试集分类准确率最高。故我们采用多项式核函数作为 SVM 核函数进行训练与动作识别。

3．训练与动作识别

下面将定义的五种人体动作分别人工标记为行走（1）、挥手（2）、拍手（3）、弯腰（4）、踢腿（5）。针对每一类动作，将提取得到的特征向量矩阵输入至多项式核函数的 SVM 中进行训练与识别分类，输出结果与分类准确率如表 6.3 所示。例如，将标记为 1 的行走动作的测试集输入到已训练好的模型中，其输出为 1 的有 873 帧，即正确识别；输出为 2,3,…,5 即为错误识别。

表 6.3　人体动作识别结果

人体动作	输出 1	输出 2	输出 3	输出 4	输出 5	准确率（%）
行走（1）	**873**	7	8	7	46	93.0
挥手（2）	13	**985**	6	0	4	98.5
拍手（3）	35	8	**886**	30	4	98.4
弯腰（4）	6	0	0	**763**	1	95.4
踢腿（5）	9	0	0	0	**795**	93.5

从表 6.3 中的结果可以看出，五种动作的识别准确率均在 90%以上，表明该算法具有很高的识别准确率，效果显著。其中，行走与踢腿动作区别性较小，因此准确率相对较低。

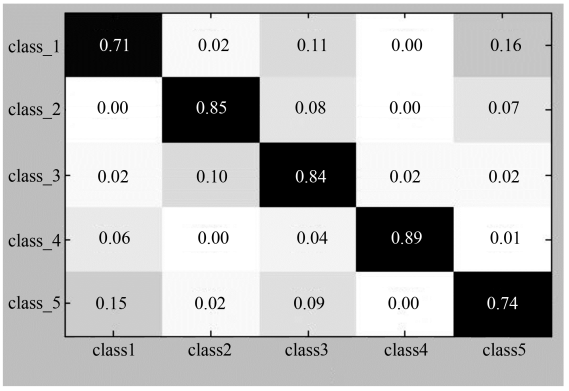
4．算法比较与分析

将本章算法与下述两种算法进行比较。

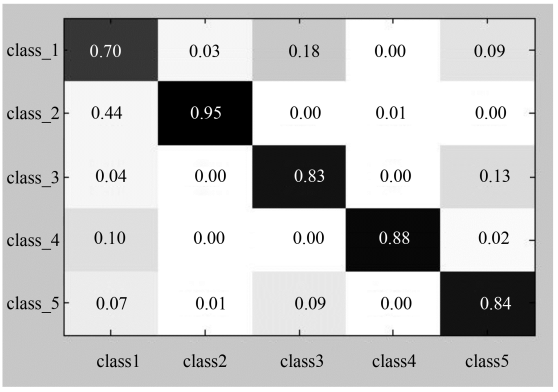
（1）算法一：基于 Kinect 直接调用骨骼节点位置信息，通过设定某一阈值判定相应动作。例如，当手掌节点位置在垂直方向上高于肩关节点位置则判定为挥手。

（2）算法二：基于运动人体轮廓特征的方法。Wang 等人的研究中提出了一种将运动人体轮廓线按照逆时针方向依次求出轮廓上的点到轮廓中心的距离形成距离向量作为特征，将该特征同样采用本章的 SVM 算法分别对本章所定义的五种动作进行训练和识别。

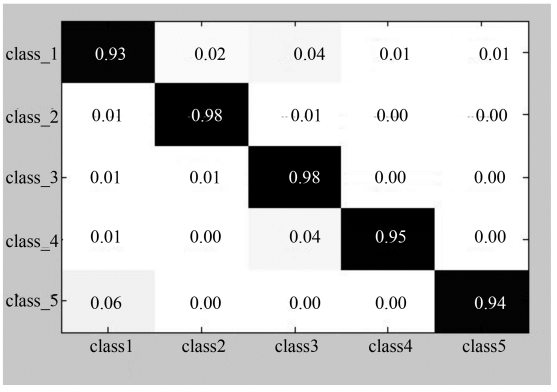
图 6.11 给出这两种算法与本章算法的识别率混淆矩阵（Confusion Matrix）的对比。



（a）算法一的混淆矩阵



（b）算法二的混淆矩阵



（c）本章算法的混淆矩阵

图 6.11　三种算法的混淆矩阵的对比

从图 6.11 的结果可以看出：

（1）算法一的平均识别准确率为 80.6%，其中行走（class1）与踢腿（class5）相似性大，导致识别率只有 70%左右，由于算法一仅通过调用 Kinect 骨骼节点的位置信息判定动作，缺少空间角度变化信息以及对数据的学习过程，这会造成待识别的动作必须要非常符合标准，否则误差很大，另一方面，对于区别性小的动作，其合理的阈值也难以确定。

（2）算法二的平均识别率为 84%，传统的提取人体轮廓特征的方法对于区别性小的、较复杂的动作识别准确率较低，轮廓特征易受噪声的干扰，而且对于动作的空间位置和角度变化而言，可靠性不高。例如，行走（class1）的识别率仅为 70%，拍手（class3）和踢腿（class5）的识别率只有 80%左右。

（3）而本章算法采用骨骼向量方向余弦这一空间角度特征，并对大量数据采用 SVM 进行训练，实现人体动作识别不受位置和空间角度变化，对于噪声干扰鲁棒性强，能够克服上述算法的缺点，其平均识别准确率达到 95.6%，实现了较为理想的动作识别结果。

6.2　基于三维时空特征的人体行为识别

本节提出了一种基于三维时空特征的人体行为识别算法。时空中梯度方向及空间角度是该算法的研究核心，首先通过引入时间维度构建三维时空概念，然后探索时空中梯度方向和空间角度信息，由梯度方向经过空间中不同的区域形成时空直方图特征来描述人体行为。

在 6.1 节中，研究了虚拟现实领域中的人体动作识别，对于人机交互系统的应用具有现实意义，它的特点是 Kinect 传感器视野是三维的，能够提供第三维深度数据，这与目前监控系统中的普通彩色视频数据不同。目前人们日常生活中，智能视频监控系统采用的都是通过普通彩色摄像头来采集数据[6-9]，因此，本节将重点对监控系统中二维视频数据的人体行为识别进行研究。

6.2.1　时空直方图特征提取

利用视频序列三维空间信息来提取有效的特征进行行为识别的研究，如今越来越受关注。这方面已有的研究例如 Laptev[10]采用 3D-Harris 的时空兴趣点提取

不同行为的时空特征；Klaser[11]的 3D-HOG 算法，它不需要进行特征点检测，而是对视频序列进行稠密采样以此提取不同行为的特征。而本章提出了一种基于三维时空特征的行为识别算法，以时空梯度方向和空间角度为研究核心。具体算法过程如下。

1．提取时空样本

我们采用的行为视频数据库是 KTH 数据库，在第 1 章介绍过，该视频数据库共有 6 种人体行为，其中每个视频数据都是同一个人在某一场景下的同一种行为，每帧图像数据的规格是（160×120）/帧。

设每个视频数据共有 a 帧图像，从中提取一段连续的 F 帧图像作为一个时空样本，为获得多个时空样本，每间隔 b 帧提取下一个时空样本。例如，第 1 个样本为第 1 帧到第 F 帧；第 2 个样本为第 1+b 帧到第 F+b 帧，以此类推，如图 6.12 所示。由此每个视频数据可以得到的时空样本数为

$$\frac{a-F}{b}+1 \tag{6.2.1}$$

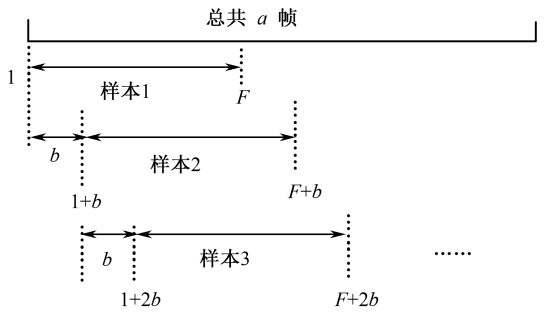


图 6.12　每个视频数据提取时空样本的方法

2．提取时空样本的时空特征矩阵

设 F=300，则每个时空样本的规格为 300 帧×（160×120）/帧。每个时空样本分别执行以下操作获取该样本的时空特征矩阵。

（1）利用采样函数对每个时空样本随机采样 p 个兴趣点，相应坐标为 (x,y,t)。

（2）针对某一个兴趣点，提取该点的特征描述符，步骤如下。

① 以某一兴趣点为中心，抽取半径为 4 像素大小的立方体。

② 将该立方体划分为固定的 8 个子立方体。

③ 每个子立方体中包含有 64 个单位立方体，每个单位立方体代表该兴趣点

领域所在尺度空间的一个像素，如图 6.13 所示。

④ 计算每个单位立方体（即一个像素点）的时空梯度方向。

⑤ 采用一个多面球体来统计每个子立方体（包含 64 个单位立方体）的时空梯度直方图。该多面球体的构造过程如下。

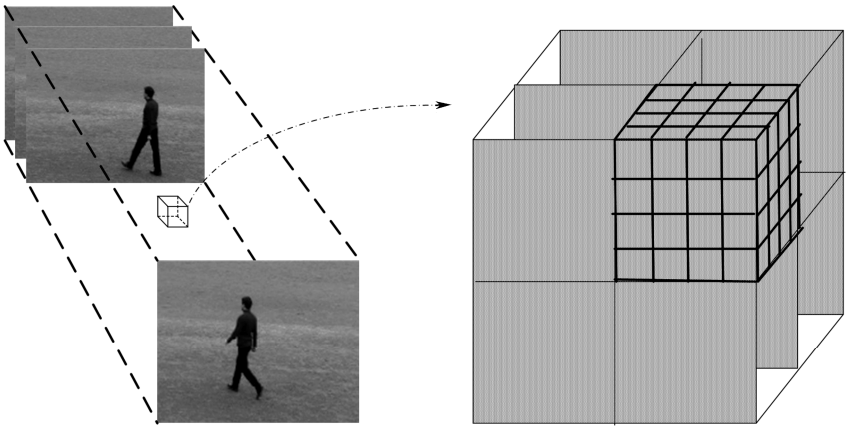


图 6.13　提取时空样本

（a）主要方法是采用“柏拉图立体”作为初始，然后进一步细化，逐渐近似球体。

（b）我们采用的是“柏拉图立体”中的“正二十面体”（它由 20 个正三角形面组成）作为初始，然后继续细化。将初始“正二十面体”的每个正三角面分成 4 个小正三角形面，如图 6.14 所示，如将面（*A*,*B*,*C*）划分为（*A*,*a*,*c*），（*B*,*b*,*a*），（*C*,*c*,*b*），（*a*,*b*,*c*）4 个小三角形面。

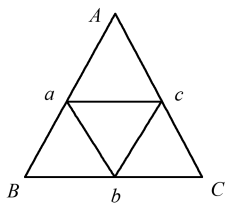
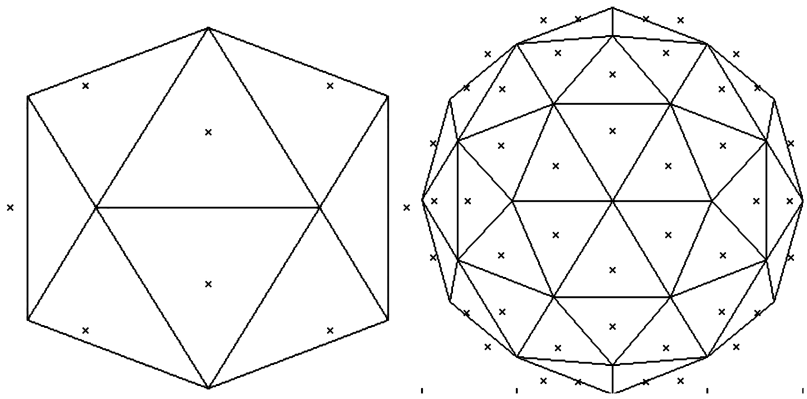


图 6.14　将初始二十面体的每个面划分为四个小三角形面

（c）由此细化后，该近似球体共有 20×4=80（个面），构成一个八十面的多面球体，如图 6.15 所示。



（a）正二十面体　　（b）细化后的八十面球体

图 6.15　多面球体

⑥ 利用该八十面球体来统计每个子立方体（即由 64 个单位立方体组成）的时空梯度方向，形成直方图。即该直方图有 80 个柱，每个柱代表正八十面体中对应的一个面区域（理由是：这样可以使得每个柱代表的时空区域大小一致），如图 6.16 所示。

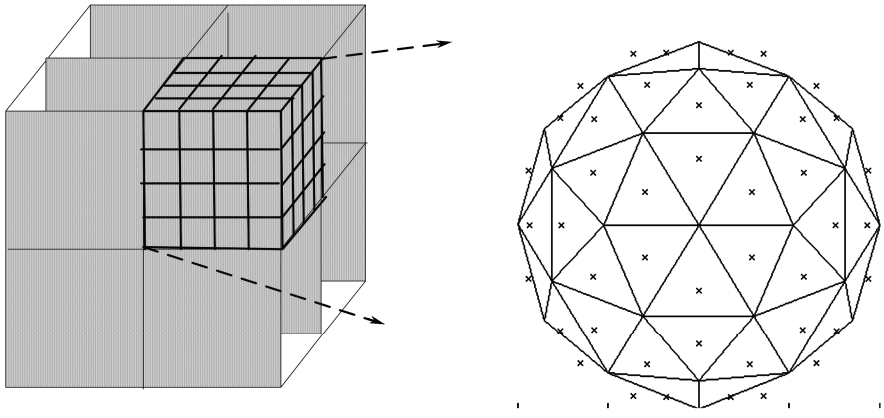


图 6.16　统计每个子立方体的时空梯度穿过多面球体相应面形成直方图

统计方法为：当某一个单位立方体的时空梯度方向穿过正八十面体中的某一个面区域时，则该面所对应的直方图的柱累加 1，由此得到一个维数为 80 的直方图。

⑦ 一共有 $2\times2\times2=8$ 个子立方体，每个子立方体得到一个维数为 80 的梯度直方图，于是该兴趣点的特征描述符的维数是 $2\times2\times2\times80=640$ 维。

（3）判定该兴趣点是否具有空间特征描述性，若没有则舍去该兴趣点，判定方法如下：

① 以兴趣点坐标为中心，抽取半径为 2 像素大小的立方体（包含 64 个单位立方体），如图 6.17 所示。

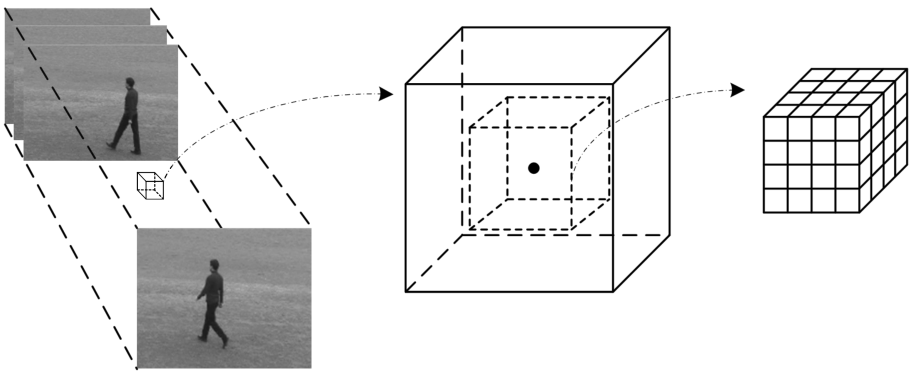


图 6.17　以兴趣点为中心抽取半径为 2 像素的立方体

② 然后依次计算每个单位立方体的梯度方向，同样按照步骤（2）中的⑥方法统计该立方体的梯度直方图，直方图维数是 80。

③ 取出该直方图中峰值 1、峰值 2 与峰值 3 的柱所对应正八十面体的相应面。

④ 计算峰值 1 所对应面的中心点 $\boldsymbol{a}$ 与峰值 2 所对应面的中心点 $\boldsymbol{b}$ 的内积，以及计算峰值 1 所对应面的中心点 $\boldsymbol{a}$ 与峰值 3 所对应面的中心点 $\boldsymbol{c}$ 的内积。公式为

$$\boldsymbol{a}\cdot\boldsymbol{b}=|\boldsymbol{a}||\boldsymbol{b}|\cos\theta_1 \tag{6.2.2}$$

$$\boldsymbol{a}\cdot\boldsymbol{c}=|\boldsymbol{a}||\boldsymbol{c}|\cos\theta_2 \tag{6.2.3}$$

因此

$$\cos\theta_1=\frac{\boldsymbol{a}\cdot\boldsymbol{b}}{|\boldsymbol{a}||\boldsymbol{b}|},\theta_1\in(0,\pi) \tag{6.2.4}$$

$$\cos\theta_2=\frac{\boldsymbol{a}\cdot\boldsymbol{c}}{|\boldsymbol{a}||\boldsymbol{c}|},\theta_2\in(0,\pi) \tag{6.2.5}$$

⑤ 设定一个合理的阈值 T_h，当 $\cos\theta_1$ 与 $\cos\theta_2$ 同时大于该阈值 T_h 时，则去除

该兴趣点，否则保留该兴趣点。

原理：若$\cos\theta_1$、$\cos\theta_2$值越大，则θ_1、θ_2角度越小，表明峰值 1 与峰值 2 落到空间区域的夹角以及峰值 1 与峰值 3 落到空间区域的夹角很小。因此，该兴趣点所在的立方体中，大多数像素点的梯度方向都穿过空间中同一小块区域，表明该兴趣点的特征区分性很小，提供的特征价值不大，所以将其去除。

（4）针对每个时空样本，经过第（2）、（3）步，去除无用的兴趣点，保留有价值的兴趣点。保留的每个兴趣点代表一个特征描述符，每个特征描述符的维数是 640 维。直至得到了 m 个特征描述符（$m < p$），则结束上述步骤。于是，我们得到了每个时空样本的时空特征矩阵为

$$X \in \Re^{m\times 640} \tag{6.2.6}$$

3．基于 K-means 的时空直方图特征提取

由于人体行为的复杂性以及外界因素的干扰，因此对于不同的人，其同种行为之间存在差异。为了能够更有效地对人体行为进行描述，本章通过采用 K-means[12]算法进一步处理所得到的时空特征矩阵，进一步提取时空直方图特征。具体做法如下。

（1）提取人体行为时空样本的时空特征矩阵。

设本章共有 M 种待识别的人体行为，其中每种行为提取 N 个时空样本，由此可以得到 MN 个时空特征矩阵，记为 $\boldsymbol{X}_j$：

$$\boldsymbol{X}_j^T = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m] \tag{6.2.7}$$

式中，$\boldsymbol{X}_j \in \Re^{m\times n}$，$n$=640，即为式（6.2.6）所述的时空特征矩阵。

（2）将上述 MN 个时空特征矩阵 $\boldsymbol{X}_j$ 用 K-means 算法聚成 K 个聚类，并求出聚类中心。

K-means 输入样本为

$$\{x_1^{(1)}, x_2^{(1)}, \cdots, x_m^{(1)}, x_1^{(2)}, x_2^{(2)}, \cdots, x_m^{(2)}, \cdots, x_1^{(MN)}, x_2^{(MN)}, \cdots, x_m^{(MN)}\} \tag{6.2.8}$$

式中，$x_i^{(j)} \in \Re^n; i \in (1, m), j \in (1, MN)$ 是一个时空特征描述符，作为一个 K-means 样本。

在 K-means 算法中，随机选取 K 个初始聚类中心，然后进行相关迭代运算，得到 K 个聚类中心，记为 $\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_k$。

（3）将 $\boldsymbol{X}_j$ 中的每个时空特征描述符 $\boldsymbol{x}_i^{(j)}$ 按上述的聚类中心进行聚类，并对 $\boldsymbol{x}_i^{(j)}$ 进行类别标记，得到 $\boldsymbol{X}_j$ 的时空特征描述聚类标记列向量 $\boldsymbol{C}_j \in \Re^m$，对 $\boldsymbol{C}_j$ 做直方图，最后得到 $\boldsymbol{X}_j$ 的时空直方图特征 $\boldsymbol{y}_j \in \Re^K$。

图 6.18 描述了上述步骤（1）～（3）的过程。

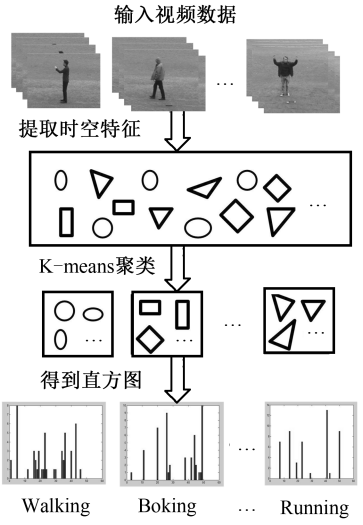


图 6.18　提取时空直方图特征步骤

（4）未知行为数据同样采用上述步骤（1）提取时空特征矩阵，然后分别求出每个时空特征描述符到 K 个聚类中心的欧式距离，距离 K 个中心中最近的则属于该类，做好标记。

（5）最后重复步骤（3），统计出每一个行为的时空直方图特征，作为测试的数据。

6.2.2　基于图像显著性的轮廓特征提取

1. 人体轮廓提取

对以往的研究可知，当人体运动时，去除背景及外界因素后，其人体轮廓蕴涵丰富的信息，从得到的轮廓中也能提取很多有用的、更高层次的特征，对进一步丰富行为描述、提高人体行为识别具有一定意义。

在前面章节中已经介绍过常用的人体轮廓提取方法。并在第 3 章中提出了一种基于图像显著性检测与背景减法线性结合的轮廓提取方法，对于某些场景下，其轮廓提取效果理想。

接下来我们通过该算法来提取人体行为数据中的人体轮廓。具体步骤如下：

（1）首先采用显著性检测算法求出输入图像的显著性值。

（2）归一化图像每个像素点的显著性值。

（3）定义一个合适的阈值。

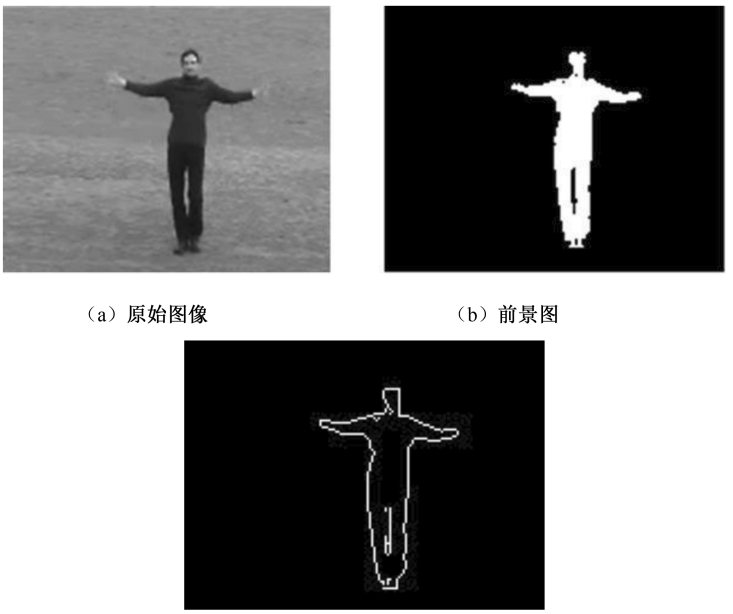
（4）比较图像中每个像素点的显著性值与阈值的大小：大于或等于阈值的点将其值置为 1；而小于阈值的点将其值置为 0，由此初步得到一幅输入图像的前景图。

（5）采用背景减法得到相应的前景图与上述前景图线性结合，设定合适的线性参数，最后得到最佳的前景图像。

（6）对得到的前景图采用膨胀与腐蚀算法进行形态学处理。

（7）最后采用 sobel 边缘检测算法处理前景图，得到最终的人体轮廓图。

图 6.19（a）为某一人体行为的原始数据图像，图 6.19（b）为采用本章的方法得到的图像前景图，图 6.19（c）为得到的人体轮廓图。



（a）原始图像　　（b）前景图

（c）轮廓图

图 6.19　提取人体行为轮廓图

2．轮廓特征提取

根据前面所得到的人体轮廓图，进一步提取其中能够描述人体运动的多种特征。为保持与 6.2.1 节算法思想一致，同样将视频数据中连续的 F 帧作为一个样本，然后分别提取该样本中每帧图像的特征，最后求出 F 帧图像的平均值作

为最后的二维特征。

本章所提取的二维轮廓特征表述如下：

（1）提取每帧图像人体轮廓的最小外接矩形的高（H）与宽（W）之比，如图 6.20 所示，最后求出每个样本连续 F 帧图像的平均值：

$$\alpha = \frac{1}{F}\sum_{i=1}^{F} H / W \tag{6.2.9}$$

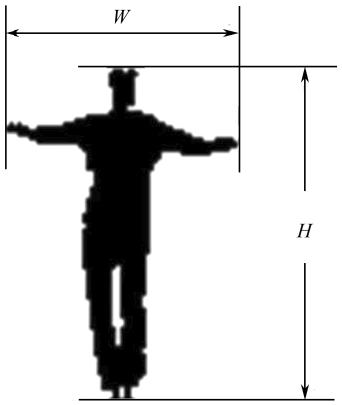


图 6.20　人体轮廓的最小外接矩形高与宽

（2）计算运动人体的姿态变化率：即前一帧的 α_{n-1} 值与当前帧的 α_n 值之比，其中 α_n 表示当前帧。最后求出每个样本连续 F 帧图像的平均值：

$$\beta = \frac{1}{F}\sum_{i=1}^{F} \alpha_{n-1} / \alpha_n \tag{6.2.10}$$

（3）提取每个样本中每帧图像的人体轮廓中心点(x,y)，然后求连续 F 帧图像的 x 与 y 的坐标值和的平均：

$$\gamma = \frac{1}{F}\sum_{i=1}^{F} (x + y) \tag{6.2.11}$$

（4）计算每帧图像中轮廓点到轮廓中心的平均距离 S，最后求每个样本 F 帧图像的平均值：

$$\delta = \frac{1}{F}\sum_{i=1}^{F} S \tag{6.2.12}$$

通过上述过程，每个样本依次可得到 4 种二维轮廓特征，定义为 $\boldsymbol{\pi}_j = [\alpha, \beta, \gamma, \delta]^{\mathrm{T}}$，把这些特征与时空直方图特征 $\boldsymbol{y}_j$ 串接起来，形成时空混合特征 $\boldsymbol{q}_j$：

$$\boldsymbol{q}_j = [\boldsymbol{y}_j^{\mathrm{T}}, \boldsymbol{\pi}_j^{\mathrm{T}}]^{\mathrm{T}} \tag{6.2.13}$$

其中，$\boldsymbol{q}_j \in \Re^{K+4}$。

6.2.3 基于 SVM 的人体行为识别

前两节中，分别得到了每个时空样本的时空直方图特征以及二维轮廓特征，并形成了时空混合特征 $\boldsymbol{q}_j$。

我们采用的数据库共有 M 种人体行为，其中每种行为提取 N 个时空样本。因此，将这 MN 个时空样本的时空混合特征 $\boldsymbol{q}_j$ 作为 SVM 训练的输入特征矩阵：

$$\boldsymbol{Q} \in \Re^{MN\times(K+4)} \tag{6.2.14}$$

矩阵 $\boldsymbol{Q}$ 的形式如图 6.21 所示。

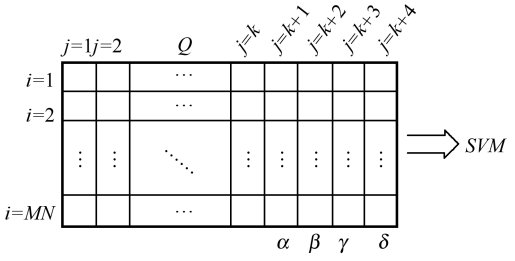


图 6.21　输入 SVM 训练的特征矩阵 $\boldsymbol{Q}$

6.2.4 行为识别结果及分析

1．训练与测试数据库

采用 KTH 行为数据库[13]用作本章算法的训练与测试。在前面章节已经介绍了，KTH 数据库是由 25 个不同性别、不同体型的人，分别进行 6 种行为（walking、boxing、hand waving、hand clapping、jogging、running），并且每个人每种行为分别在四种场景（户外、户外镜头变焦、户外不同着装、室内）下依次采集，图 6.22 所示为部分示例。

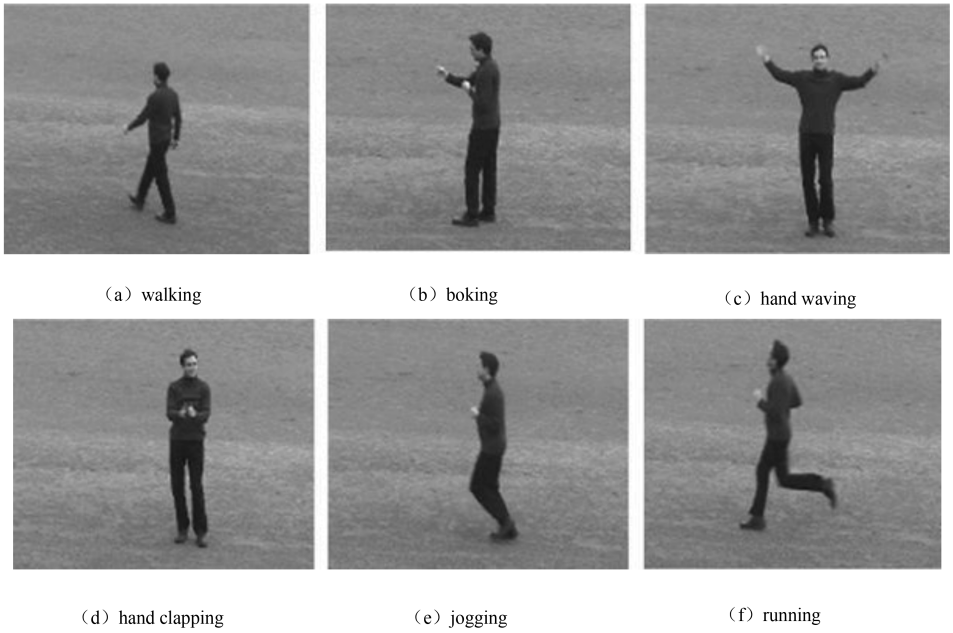


图 6.22　KTH 行为数据库示例

2．识别结果分析

（1）本实验采用每隔 10 帧提取连续的 300 帧作为一个时空样本，即 b=10，F=300 提取每个视频数据中的时空样本。得到的时空特征矩阵 $\boldsymbol{X}_j$ 的维数为：m=50，n=640。

（2）待识别的行为类数为 M=6，分别人工标记为：walking（1）、boxing（2）、hand waving（3）、handc lapping（4）、jogging（5）、running（6）。每种行为提取时空样本数为 N=300，采用 K-means 聚类数为 K=150。

（3）我们采用每种行为的测试样本数为 100，6 种行为的测试样本数总计为 600 个。

图 6.23 为上述参数所得到的六种行为的识别率混淆矩阵：

从图 6.23 中的结果可以看出，这六种行为的识别准确率均在 90%以上，表明本章算法效果显著。其中 walking、jogging、running 这三种行为区别性较小，因此准确率相对较低。

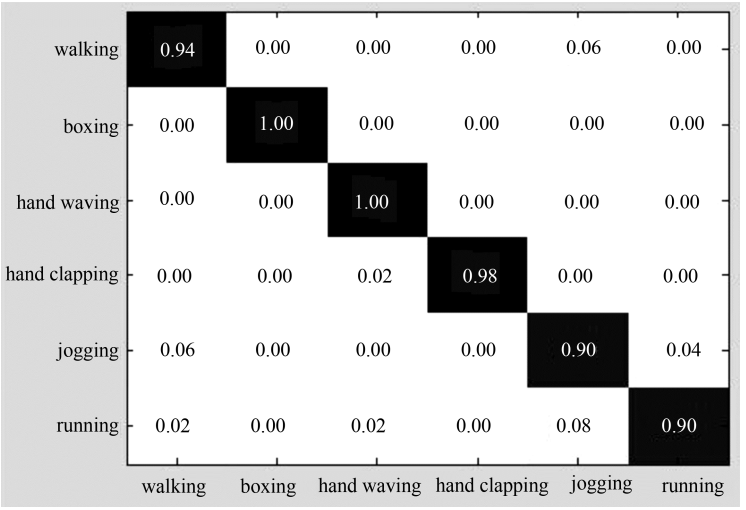


图 6.23　K=150、N=300 时的识别混淆矩阵

3．不同参数值选取对识别率的影响

当 K-means 聚类数 K 取不同值，以及输入 SVM 每种行为的时空样本数 N 取不同值时（图中两条曲线分别代表六种行为的总时空样本为 1800 和 1200），其平均识别准确率的变化曲线如图 6.24 所示。

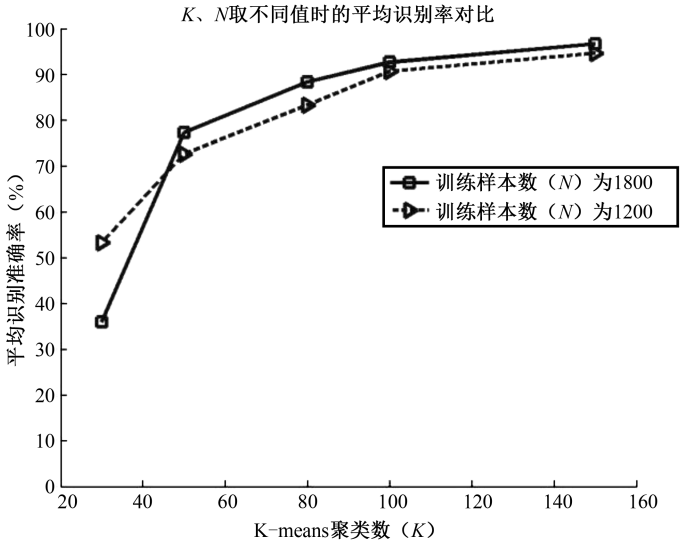


图 6.24　K、N 取不同值时的平均识别率曲线

从图 6.24 中可以看出：①当 K-means 聚类数低于 50 时，识别效果很差，而且此时增加时空样本数也无法提高识别率；②而随着聚类数 K 和时空样本数 N 的增加，平均识别率显著提高，当 K=100 时，准确率能达到 90%左右；③但是当聚类数和时空样本数达到一定值时，识别率曲线趋于平缓，此时很难再提高准确率，反而当 K 与 N 的值过高时，将会大大增加算法的时间成本，因此需要在耗费时间与追求准确率之间寻找适合的平衡点。

4．不同算法的识别率对比

为了反映本章算法的有效性，将本章算法与其他算法对 KTH 行为数据库的识别准确率进行对比，如表 6.4 所示。

表 6.4　不同算法对 KTH 行为数据库的识别率对比

算　　法	识别准确率（%）
算法一：仅采用二维轮廓特征	85.76
算法二：3D Space-Time Interest Points[14]	93.17
算法三：Dense Trajectories[15]	94.20
本章算法	95.33

注：表中算法一采用的是 6.2.2 节中所述的二维轮廓特征 $\boldsymbol{\pi}_j$ 。

从表中的数据可以看出本章算法有很高的识别准确率。

算法一：仅采用二维轮廓特征，其识别率不高，因为传统的提取人体轮廓特征的方法对于较复杂的行为而言，易受噪声的干扰，而且对于行为的空间位置和角度变化而言，可靠性不高。

算法二：Bregonzio 等人[14]采用 3D 时空兴趣点检测算法，其识别率达到 93.17%。

算法三：Wang 等人[15]通过采样图像稠密点并追踪其轨迹，以此形成稠密点特征轨迹来识别人体动作，其识别率达到 94.20%，这两种算法都实现了较好的识别效果。而本章算法从两个角度出发，同时利用三维时空特征和二维轮廓特征来描述人体行为，实现了较为理想的识别结果。

6.3 本章小结

本章提出了一种基于空间几何角度算法来实现人体动作的识别。该研究着眼于虚拟现实、人机交互中的人体动作识别，通过采用 Kinect 深度传感器获取人体 20 个骨骼关节点。然后进一步提取人体相应动作的骨骼空间几何角度特征：一类是骨骼空间方向余弦特征，一类是关节点连接相邻的两骨骼之间的空间夹角特征。

与传统人体动作识别的方法不同，本章的方法不受光线、位置等外界因素的影响，另一方面，传统的方法需要采用复杂的数学算法，造成实现困难以及运行效率不高，而本章以简单直观的方法，直接研究人体骨骼在空间中的几何角度信息，在较大样本的训练下对人体不同动作的识别准确率均达到 90%以上，实验结果表明该方法具有很好的有效性和鲁棒性。

同时，也重点提出了一种时空直方图的特征提取方法，以时空梯度方向和空间角度信息为研究核心。该研究焦点是对监控系统中普通视频数据的人体行为识别研究，与以往提取人体行为二维静态特征算法不同，本章算法的核心是引入了第三维时间数据，并探索时空中梯度方向以及空间角度来描述人体行为。

二维轮廓特征对于复杂的行为而言，有效性不高，考虑到时空中蕴含着丰富的能够描述行为关键位置以及空间角度的信息，因此将两者结合能够实现更为丰富和完整的描述人体行为。在 6.2.1 节中重点研究了提取时空直方图特征的过程，6.2.2 节探索多种二维轮廓特征的描述，6.2.3 节中结合以上所获特征通过支持向量机进行训练与识别，6.2.4 节从多个角度以及算法对比方面分析了本章方案的识别效率。

本章参考文献

[1] 赵沁平. 虚拟现实综述[J]. 中国科学, 2009, 39(1): 2-46.

[2] Mueller M, Karasev P, Kolesov I, et al.. Optical Flow Estimation for Flame Detection in Videos[J]. IEEE Transactions on Image Processing, 2013, 22(7): 2786-2797.

[3] Han J, Shao L, Xu D, et al.. Enhanced Computer Vision with Microsoft Kinect Sensor: A Review[J]. IEEE Transactions on Cybernetics, 2013, 43(5): 1318-1334.

[4] Raptis M, Kirovski D, Hoppe H. Real-time classification of dance gestures from skeleton animation[J]. ACM SIGGRAPH/Eurographics Symp. Comput. Animation, 2011: 147-156.

[5] Cortes C, Vapnik V. Support-vector networks[J]. Machine Learning, 1995.09, 20(3): 273-297.

[6] Zhou Feng, Torre F D, Hodgins J K. Hierarchical Aligned Cluster Analysis for Temporal Clustering of Human Motion[J]. IEEE Transactions on PAMI, 2013, 35(3): 582-596.

[7] Ramanathan M, Yau Wei-yun, Teoh E K. Human Action Recognition With Video Data: Research and Evaluation Challenges[J]. IEEE Transactions on Human-Machine Systems, 2014, 44(5): 650-663.

[8] Everts I, Gemert J C, Gevers T. Evaluation of Color Spatio-Temporal Interest Points for Human Action Recognition[J]. IEEE Transactions on Image Processing, 2014, 23(4): 1569-1580.

[9] Popoola O P, Kejun Wang. Video-Based Abnormal Human Behavior Recognition—A Review[J]. IEEE Transactions on Systems, Man, and Cybernetics, 2012, 42(6): 865-878.

[10] Laptev I, Lindeberg T. Space-time Interest Points. Ninth IEEE International Conference on Computer Vision[C]. France: IEEE, 2003, 13-16.

[11] Klaser A, Marszalek M, Schmid C, et al.. A Spatio-Temporal Descriptor Based on 3D-Gradients[C]. 19th British Conference on Machine Vision, British, 2008: 1-10.

[12] Krishna K, Murty M N. Genetic K-means Algorithm[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics. 1999, 29(3): 433-439.

[13] Schuldt C, Laptev I, Caputo B. Recognizing human action: a local SVM approach. Proceedings of the 17th International Conference on Pattern Recognition [C]. Sweden: IEEE, 2004:32-36.

[14] Bregonzio M, Gong Shao-gang, Xiang Tao. Recognising Action as Clouds of Space-Time Interest Points. IEEE Conference on CVPR[C]. Miami, FL: IEEE, 2009, 1948-1955.

[15] Heng Wang, Klaser A, Schmid C, et al.. Action Recognition by Dense Trajectories. IEEE Conference on CVPR[C]. Providence RI: IEEE, 2011, 3169-3176.

第 7 章

Kinect 应用示例

通过上一章的介绍，我们对 Kinect 深度传感器有了较为深入的了解。鉴于 Kinect 深度传感器应用的日益广泛，本章将通过两个应用案例进一步地介绍它的应用。

7.1 基于深度信息的手势识别的实现

本章针对复杂环境下的手势识别问题，提出了一种基于深度图像的静态手势识别系统。目标是该系统可以识别出预先设定的数字手势，即“1”、“2”、“3”、“4”、“5”五种手势。预先设定的手势如图 7.1 所示，该系统的功能框图如图 7.2 所示。

图 7.1　预先设定的手势

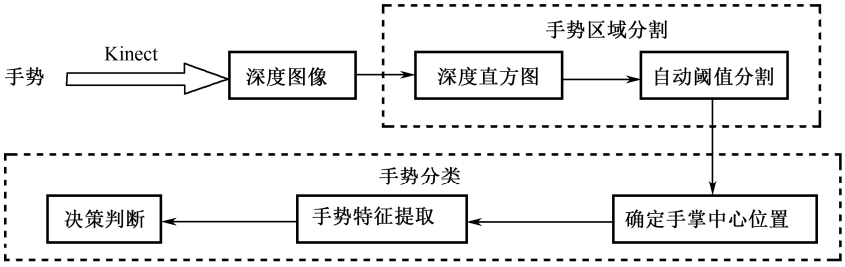


图 7.2　基于深度图像的手势识别系统框图

系统工作流程简述：

（1）该系统首先通过 Kinect 的深度摄像头获取到场景的深度图像。

（2）手势区域分割。

场景中除了手部以外还有许多冗余的干扰信息，所以下一步是把要用于手势识别的手掌部分分离出来，这称为手势区域分割[1]。在进行手势区域分割时，首先生成关于场景深度图像的深度统计直方图。深度直方图记录了深度图像在各个距离区间上的像素点数，根据直方图的统计特性，可以找到一个阈值并且分离出手势区域。这个阈值又与每个场景中手掌到摄像头的距离有关，所以每次可能各不相同。为此，本章设计了一种自动阈值技术来取得每个场景下的最优阈值。深度图像中满足阈值条件的像素点的集合就是手势区域（又称为“前景”），同时将深度图像二值化，前景像素点的数值设为“1”，其他的设为“0”。

（3）手势分类。把手势区域分离出来之后，就可以进行手势分类了。本章定义的 5 种手势的区别就是每种手势伸出的手指数量不一样，所以手势识别模块的最终目的就是计算出手指的数量，来判断手势。设计的算法的核心思想可以简单归结为：以手掌为圆心，绘制多个半径不同的同心圆，分析计算得出每个同心圆上有几根手指，之后再判断出一共伸出了几根手指，这样就能识别出对应的手势。为此，第一步就是要在分离出的手势区域中找到手掌的中心。在这里使用了数学形态学的图像腐蚀技术找到中心。对于手势特征的提取，设计并实现了一种可称为“画圈圈，数手指”的方法。使用该方法，可以找出伸出的手指的数量。最后，通过手指的数量来判断是哪个手势。

7.1.1　基于 Kinect 的深度信息的获取

微软的 SDK 中提供了获取深度图像的 API。在初始化设置之后，就可以从深度数据流中读取深度信息。获取的深度图像的大小是宽 640 像素×高 480 像素。

Kinect 获取的深度图像的每个像素点由 2 个字节组成，共 16 位。其中，高十三位代表从红外摄像头到物体的距离，以毫米为单位。低三位为被跟踪的用户索引号，如果检测到用户身体，就会自动分配一个编号。

高十三位表示的距离区间为 0 到 4096mm。因为该算法不需要很高的精度，仅仅把 13 位的距离信息转换为 8 位数据存储。转换时使用等比例缩放，丢弃高精度信息，得到深度图像。然后使用 OpenCV 的 IplImage 结构来存储该图像。图 7.3 和图 7.4 为所得到的彩色图像和深度图像。

图 7.3　彩色图像

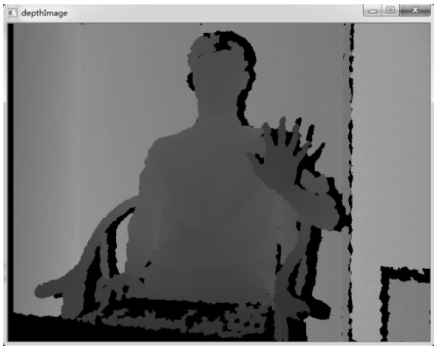


图 7.4　深度图像

【编程简析】

实现从 Kinect 深度数据流中得到深度图像的函数为：

void getDepthImage(HANDLE &depthEvent, HANDLE &depthStreamHandle, IplImage *depthImage)

参数：depthEvent，depthStreamHandle 为 Kinect 深度数据流的事件监听器，depthImage 为得到的深度图像。

……

```
int data = (bufferRun[j]&0xfff8) >> 3;  //读取出深度数据流的高 13 位
ptr[j] = (uchar)(256*data/0x0fff);          //将 13 位数据转换为 8 位
```

……

7.1.2　手部区域分割

基于深度信息的手势识别与传统的基于光学信息的手势识别的最大区别就是手部区域的分割。相比而言，基于深度信息的区域分割更加方便。获取到深度图像之后，下一步就是把手掌区域从环境中分离出来，为手势识别做准备。本章

提出的手部区域分割是建立在一个前提假设之上的：手掌区域是距离摄像头最近的区域。人与人交流时，如果要有手势的交流，通常情况下都会把手伸到身体的前方做手势。那么，在人机交互的时候，使用者会很自然地将手伸到身体前方做手势。这时，手掌区域一般会是距离摄像头最近的区域。所以，这个假设是合理的。有了这个假设之后，分离出手掌区域就变成了一件比较简单的工作，即找到距离最近的像素点的集合。为了完成该目的，使用深度直方图[2]。通过它的统计特性，就可以自适应地确定分割图像所需的阈值。

1．深度直方图

直方图（Histogram）是一种二维统计图表，它的两个坐标分别是统计样本和该样本对应的某个属性的度量。它广泛应用于很多计算机视觉应用中。在这里我们关心的是深度图像中深度值在距离区间上的分布。直方图能够直观地反映给定数据集中数据的分布状况。从直方图中，我们能够看出深度值出现的频率以及聚集分组。通过这些信息，我们能够确定阈值以及其他能够用来对图像进行过滤的指标，使得能够最大化地揭示深度影像图中的深度信息[3]。

这里，使用的直方图横坐标为距离区间，纵坐标为该距离区间上的像素点的数量。图 7.6 为图 7.5 所对应的深度直方图。

图 7.5　深度图像

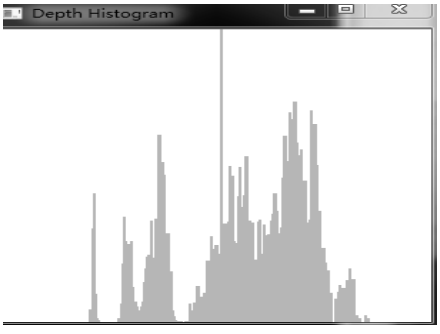


图 7.6　深度直方图

【编程简析】

实现从深度图像中计算得到深度直方图的函数为：

```
void processDepthHistogram(IplImage *src,IplImage* dst)
```

参数：src 为输入的深度图像，dst 为输出的深度直方图。

```
……
//直方图初始化
for (i=0;i<HN;i++) hist[i] = 0;
```

```
//遍历图像，计算直方图
for(i=0;i<height;i++)     //行
{
      for(int j=0;j<width;j++)    //列
      {
      hInterval = data[i*step+j]/hStep;    //计算该点距离属于的距离区间号
      hist[hInterval]++;                   //对应距离区间点数增加
       }
}
……
```

2．自动阈值分割

观察图 7.6 可以发现，深度图像的像素点主要集中在三个距离区间上，称为出现了三个聚簇。大部分的像素点是在最右边的聚簇上，这些像素点对应的是场景中距离最远的背景环境。这部分对于手势识别没有帮助，需要舍弃。中间聚簇的像素点主要是人体躯干对应的位置区间，这也是需要舍弃的。而最左侧的聚簇对应的像素点就是我们感兴趣的距离最近的手掌区域。我们的目标就是把这部分像素点分割出来。也就是说，我们可以通过找到这个聚簇与旁边聚簇的分割点来确定该聚簇的距离范围，而这个分割点就是阈值。通过实验发现，当手掌未伸出或者手掌很贴近身体躯干时，三个聚簇会变为两个，如图 7.7 与图 7.8 所示。此时，无法分割手部区域。正常情况下，我们假定手部区域在身体的前方。接下来就是要确定分割阈值。图 7.6 中，左侧聚簇右端有几个距离区间的数值为零。一开始，尝试通过找到这个中零区间来确定阈值，但实验发现，这种方法不能很好地实现手部分割。图 7.9 中的直方图也满足了我们的假设，但是就没有中零区间。

图 7.7　手掌未前伸的深度图像

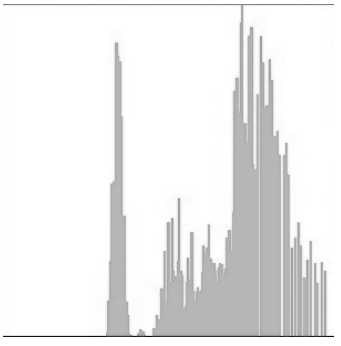


图 7.8　手掌未前伸的深度直方图

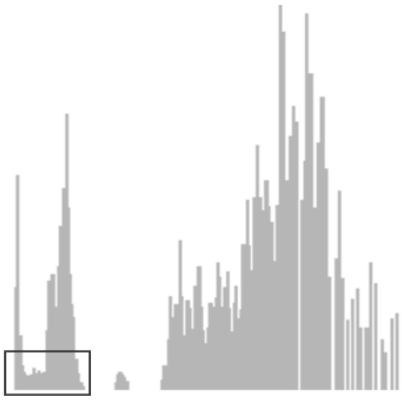


图 7.9　没有中零区间的深度直方图

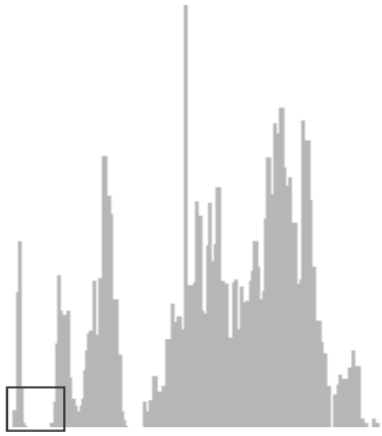


图 7.10　有中零区间的深度直方图

通过实验发现，深度直方图中“中零区间”的有无与使用者胳膊的位置有关。如果手掌与胳臂垂直并且在深度摄像头的视角上遮挡了胳膊，则直方图中就会出现中零区间，如图 7.10 所示。反之，手掌并未遮挡住胳膊时，则不会出现中零区间。通过多次试验观察发现：只要满足了我们的前提假设，无论手掌是否遮挡住了胳膊，都会出现一个峰值（对应手掌区域），并且在峰值的右侧都会出现一个高度小于峰值的一半的区间。而右侧最近的小于半峰值的距离区间对应的就是手腕与胳膊的接合处。实验证明，该区间的起始位置是分割手掌区域的最优阈值，如图 7.11 所示。

图 7.11　深度直方图中的最优阈值（红色标记的区间）

找到阈值之后，下一步是手掌区域的分割。深度图像中，距离小于阈值的所有的像素点判定为手掌区域的像素点，并且将这些像素点的数值改为“1”；大于阈值的像素点则被舍弃，将数值改为“0”。并且记录下手掌区域的边界位置。这样，就从深度图像中分离出了手掌区域，并且完成了图像的二值化。图 7.13 为从图 7.12 中分割出的手掌。

然而，通过实验发现当手臂与手掌在同一距离平面上时，分割出的手部区域

会带有手臂。为了增加算法鲁棒性，为前景区域增加了一个高度限制——前景区域的高度不能超过深度图像高度的 3/10。如此，算法很好地避免了手掌未充分前伸时手势分类错误的发生。

图 7.12　场景深度图像

图 7.13　分割出的前景手掌图像

【编程简析】

实现从深度直方图自动确定阈值并且分割图像的函数为：

bool extractForeground(IplImage *depthImage,IplImage* forground)

参数：depthImage 为输入的深度图像，forground 为分割完成的前景图像。

……

```
//确定前后阈值
    for(i=0;i<HN;i++)
    {
        if(hist[i] == 0) nearThreshold = i*hStep;
    //最小深度阈值为第一个非零距离区间的起始深度
    if((i>4)&&(hist[i]<hist[i-1]/2)&&(hist[i]<hist[i-2]/2)&&(hist[i]<hist[i-3]/2)))
        {
    //自动阈值技术决定最大深度阈值，条件为当前距离区间的点数小于前几个区间的点数
            farThreshold = i*hStep;
            break;
        }
    }
```

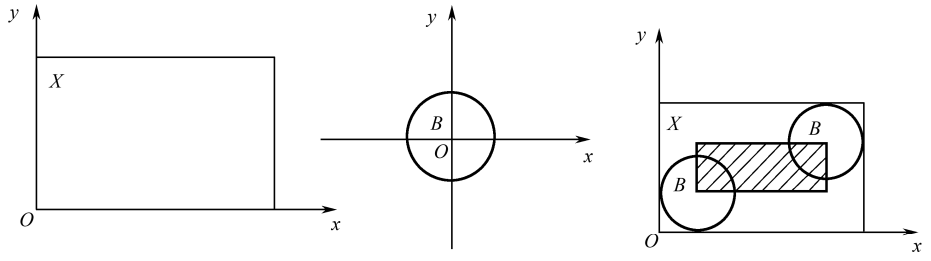
……

7.1.3　手势分类

手势区域分割实现了从复杂的背景中分离出了需要识别的手掌区域，下一步就可以进行手势分类了。预先定义的 5 种数字手势的区别是伸出的手指数量不一样，通过这个区别就可以判定所对应的手势。这里使用的算法的核心思想可以简单归结为：以手掌为圆心，绘制多个半径不同的同心圆，分析计算得出每个同心圆上有几根手指，之后再判断出一共伸出了几根手指，这样就能识别出对应的手势。为此，第一步就是要在分离出的手势区域中找到手掌的中心。之后，进行手势特征的提取。设计并实现了一种名为“画圈圈，数手指”的方法。使用该方法，可以找出伸出的手指头的数量。最后，通过手指的数量来判断是哪个手势。

1．手掌中心定位

分割出的前景如图 7.15 所示，除了手掌以外还可能有几个伸出的手指。所以前景区域的中心位置不一定是手掌的中心位置。为了找到手掌的中心位置，使用图像形态学中的图像腐蚀技术来找到手掌的中心。图像腐蚀处理可以表示为用结构元素对图像进行探测，找出可以放下该结构元素的区域。如图 7.14 所示，左侧为原始图像，X 为原始图像中为“1”的像素点的集合。中间的 B 为腐蚀使用的结构元，B 的圆心为结构元素的中心。右侧中阴影区域为腐蚀的结果。腐蚀的工作流程为：将结构元素 B 在 X 所在的整个图片中移动。当 B 的所有的点都包含在 X 中时，记录下 B 的中心点。所有满足该条件的中心点的集合就是腐蚀的结果，如图 7.14（c）的阴影区域。



（a）原始图像　　（b）结构元素　　（c）腐蚀结果

图 7.14　图像腐蚀

通过实验发现，对于带有手指的前景图片，使用正方形当做图像腐蚀的结构

元素是效果最好的。并且通过实验确定了结构元素的最优尺寸。图 7.16 为图 7.15 的图像腐蚀结果。通过找到图 7.16 中白色区域的中心位置，就确定了前景手掌的中心位置。

图 7.15　手部前景图像

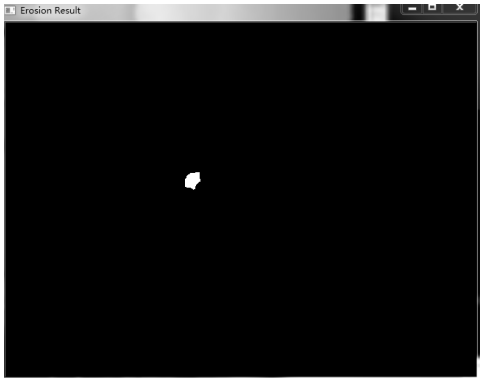


图 7.16　图像腐蚀结果

【编程简析】

实现将二值化前景图像进行腐蚀得到手掌中心的函数为：

bool erosionAndFindCenter(IplImage *src,IplImage *dst)

参数：src 为输入的前景图片，dst 为输出的腐蚀结果。

……

```
IplConvKernel *element = cvCreateStructuringElementEx(an*2+1,
                        an*2+1,an,an,CV_SHAPE_RECT,0);//创建腐蚀结构元
cvErode(src,dst,element,1);//腐蚀图像
for (int i=0; i<dHeight; i++)
    {
        uchar *ptr      = (uchar*)(dst->imageData+i*dst->widthStep);
        for (int j=0; j<dWidth; j++)
        {
            if(ptr[j] != 0)
            {
            //找到腐蚀结果边界
                if(upY>i)       upY=i;
                if(downY<i)   downY=i;
                if(leftX>j)           leftX=j;
                if(rightX<j)    rightX=j;
```

```
                }
            }
        }
//手掌中心为腐蚀结果边界的中心
palmCenter.x  =(leftX+rightX)/2;
palmCenter.y  =(upY+downY)/2;
……
```

2．手势特征提取与决策

事先设定的五种手势的区别就是伸出的手指的数量，所以手势的特征就是手指的数量。对于特征提取，使用“画圈圈数手指”法。如图 7.17 所示，该方法的工作流程如下：

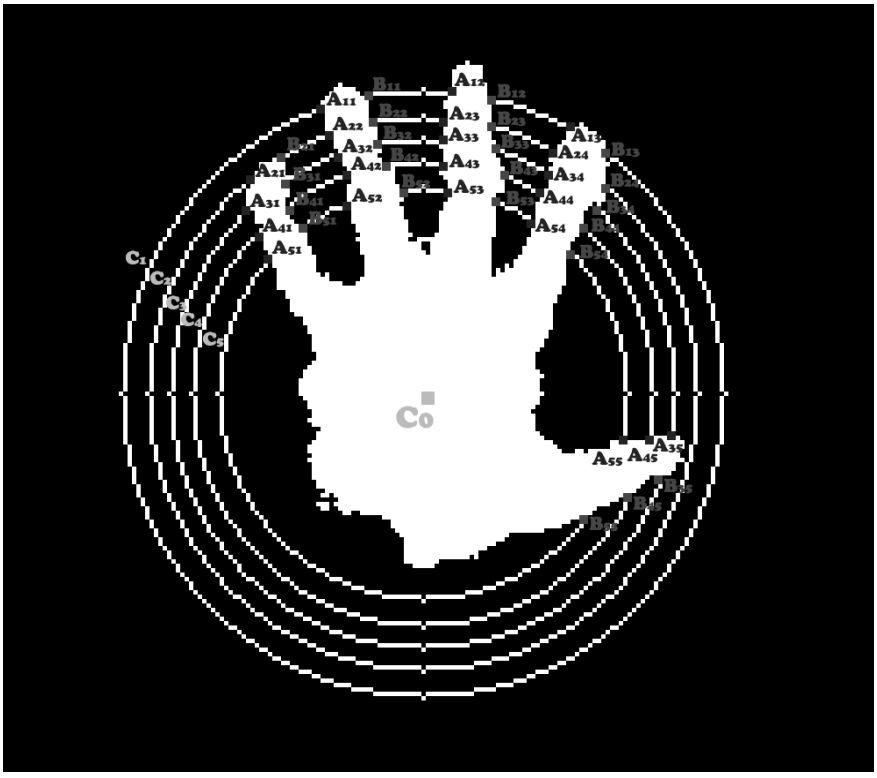


图 7.17　画圈数手指

（1）通过之前获得的手掌中心 C0 和前景区域坐标，计算出 C0 距离前景区

域边界的最大距离 L。

（2）以 C0 为圆心，$r=i\times L/N$ 为半径绘制多个同心圆。N 为定值，i 取（N−1, N−2, N−3, …, N/2)。图 7.17 中的 C1、C2、C3、C4、C5 为按此方法绘制出的同心圆。

（3）按顺时针方向遍历同心圆上的所有的插值像素点，判断是否与上一个像素点的数值有变化。如果上一个像素点为“0”（图 7.15 中的黑颜色像素点），而当前像素点为“1”（白颜色像素点），则当前像素点为 A 点。若上一个像素点为“1”，而当前像素点为“0”，则当前像素点为 B 点。记录同心圆上的 A 点与 B 点的数量，并编号。如上图 C1 同心圆上，从手掌左侧开始遍历，首先出现了一个 A 点，该点为“1”且上一点为“0”，则将其编号为 A11。下标的第一位代表同心圆的编号，第二位代表 A 点的出现次序。继续遍历，出现了一个从“1”变为“0”的 B 点，编号为 B11。同理，得到该圆上的 A12、A13、B12、B13。按照同样的方法处理剩下的同心圆，则可得到图 7.17 中的 A 点和 B 点。

（4）对于每一根伸出的手指，如果出现了一个 A 点，则必然出现一个 B 点，即 A 与 B 是成对出现的。如果出现单独的 A 点或者 B 点，则对应的位置不是一根完整的手指，对结果有不好的影响。所以，为了鲁棒性，丢弃单独出现的 A 点或者 B 点。

（5）统计每个同心圆上 A 点与 B 点的对数。所有同心圆中，最大的对数就是该手势伸出手指的数量，也就是该手势的特征。如图 7.15 中，C1 对数为 3，C2 对数为 4，C3 对数为 5，C4 对数为 5，C5 对数为 5。最大的对数为 5，所以该手势的特征值为 5，即伸出了 5 根手指。

（6）将手势的特征值对应为手势。特征值为 1，则为 1 号手势。特征值为 2，则为 2 号手势。以此类推。

【编程简析】

实现手势分类的函数为：

```
int countingFingers(IplImage* forground)
```

参数：forground 为输入的前景二值化图像。

……

```
//中心到左上角距离
maxDistance  = sqrt(float((palmCenter.x - foreROI.x)*(palmCenter.x - foreROI.x)
                          + (palmCenter.y - foreROI.y)*(palmCenter.y - foreROI.y)));
//中心到右下角距离
tempDistance  = sqrt(float((foreROI.x + foreROI.width - palmCenter.x)*(foreROI.x +
foreROI.width - palmCenter.x)+ (foreROI.y + foreROI.height -
```

```
palmCenter.y)*(foreROI.y + foreROI.height - palmCenter.y)));
//边界距离为到左上角或右下角中较远的距离
if(tempDistance>maxDistance)     maxDistance = tempDistance;
data =(uchar *)forground->imageData;
numOfFingers = 0;
    //按不同的半径绘制圆圈
    for(i=circleN-ignoreOuterCN;i>ignoreInerCN;i--)
    {
        //(lX,lY)为上一个像素点;(cX,cY)为当前像素点
        cRadius = (i+1)*maxDistance/circleN;        //当前圆圈的半径
        lX = int(C0x + maxDistance);
        lY = C0y;
        //遍历某个圆圈上的所有插值像素点
        for (j=0;j<numPointOnCircle;j++)
        {
            phase     = j*2*PI/numPointOnCircle;
            cX        = int(C0x + cRadius* cos(phase));
            cY        = int(C0y + cRadius* sin(phase));
            if(cX==lX && cY==lY)continue;
            lData     = data[lY* forground->widthStep +lX];
            cData     = data[cY* forground->widthStep +cX];
            //如当前像素点的数据与上一个像素点数据有变化，则出现变化奇点
            if(lData != cData)
            {
                if(cData == 0) numCPwhite2black[i]++;      //白变黑点
                else numCPblack2white[i]++;                //黑变白点
            }

            lX        = cX;
            lY        = cY;

        }
        numCP[i] = (numCPwhite2black[i] < numCPblack2white[i]) ?
numCPwhite2black[i] :   numCPblack2white[i]; //舍去单独的白变黑点或黑变白点
```

```
            //手指数量为所有圈中最大的颜色变化奇点数量
            if(numCP[i] > numOfFingers)    numOfFingers = numCP[i];
            cvCircle(forground,cvPoint(C0x,C0y),cRadius,cvScalar(255));
        }
……
```

7.1.4 实验结果

实验软件环境：Microsoft Visual Studio 2010 旗舰版，Microsoft Kinect Windows SDK v1.7，OpenCV 2.4.5。

硬件环境：Kinect for Xbox 360, 2010 年款 15 寸 Macbook Pro（CPU，Intel M620 双核四线程 2.66GHz；内存，8GB）。

编程实现该算法后，经过测试，证明该算法可以成功地实现预定的目标，即以每秒 30 帧的速率检测出场景中使用者的手势。即使当使用者的手掌以不同的角度旋转时，仍可以成功地完成手势识别。并且本系统能够在各种光照条件下正常工作。

1. 正常使用条件下的实验结果

正常使用条件是指使用环境的光照条件为自然光，使用者前伸手掌，手掌平面与传感器平面平行，手指方向为向上。实验结果如图 7.18～图 7.22 所示。

图 7.18　成功识别 1 号手势

图 7.19　成功识别 2 号手势

图 7.20　成功识别 3 号

图 7.21　成功识别 4 号手势

图 7.22　成功识别 5 号手势

正常使用条件下测试数据如表 7.1 所示。

表 7.1　算法测试结果数据

手势编号	测试次数	成功识别次数	手势识别率
5	500	498	99.6%
4	500	494	98.8%
3	500	496	99.2%
2	500	479	95.8%
1	500	488	97.6%

可以看出，在正常使用条件下本系统有很高的成功识别率。即使是识别效果最差的数字手势“2”也有高达 95.8%的识别率。

2．同光照条件对实验结果的影响

传统的基于光学信息的手势识别的最大限制就是对于使用环境的光照条件有很苛刻的要求。而本系统采用的是深度图像，它是通过主动投射近红外线后成像而来的。所以本系统对于光照条件基本没有要求。为了验证该点，本章进行了相关测试。实验结果证明本系统可以适应变化的光照条件。图 7.23 是正常光照条件下本系统的结果，图 7.24 是低光照条件下本系统的结果，表 7.2 则为不同光照条件下对于手势 4 的识别结果。

表 7.2　不同光照条件下的实验结果

光照条件	测试次数	成功识别次数	手势识别率
全黑暗条件	200	197	98.5%
低光照（光源只有电脑屏幕）	200	195	97.5%
正常光照（自然光）	200	196	98.0%
正常光照（白炽灯）	200	198	99.0%
强光照（自然光加白炽灯）	200	194	97.0%

图 7.23　正常光照条件下成功识别手势 4

图 7.24　低光照条件下成功识别手势 4

3．手掌旋转对实验结果的影响

正常使用条件下，手心正对传感器，手指方向为正上方。手掌旋转是指手指方向与正上方呈一定夹角。因为本系统是靠手指的数量进行分类，所以手掌的旋转对于实验结果基本没有影响。同理。手掌的翻转也对结果没有影响。表 7.3 为手势 4 旋转后的测试结果，图 7.25 和图 7.26 为旋转后的软件截图。

表 7.3　手势 4 旋转后的测试结果

旋转角度	测试次数	成功识别次数	手势识别率
顺时针旋转 90°	200	197	98.5%
顺时针旋转 180°	200	196	98.0%
顺时针旋转 270°	200	196	98.0%

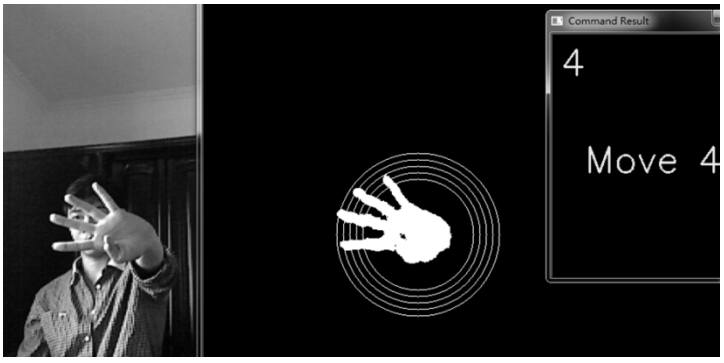


图 7.25　手掌逆时针旋转 90° 后仍能正确识别

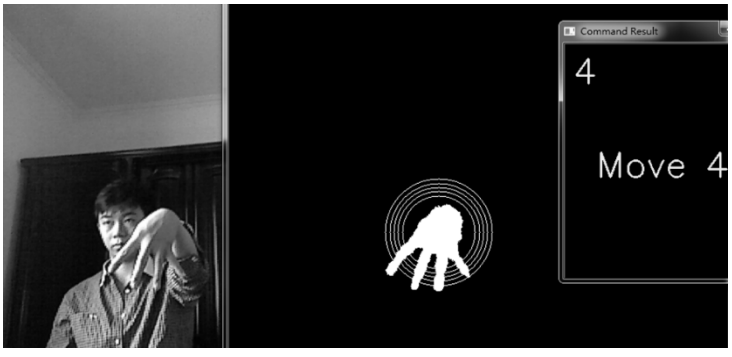


图 7.26　手掌逆时针旋转 180° 并翻转后仍能正确识别

4．手掌倾斜对实验结果的影响

通过对于本系统的测试发现，手掌的倾斜对于手势识别准确率的影响是最大的。倾斜是指手掌平面与传感器平面不平行而成一夹角，如图 7.27 和图 7.28 所示。

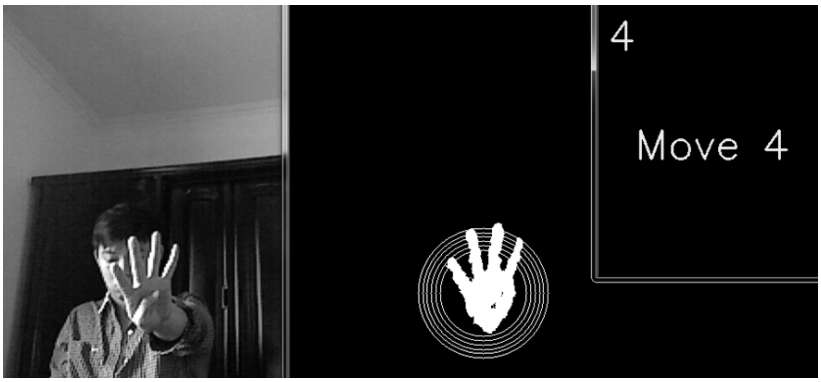


图 7.27　手掌倾斜 30° 后仍正确识别手势

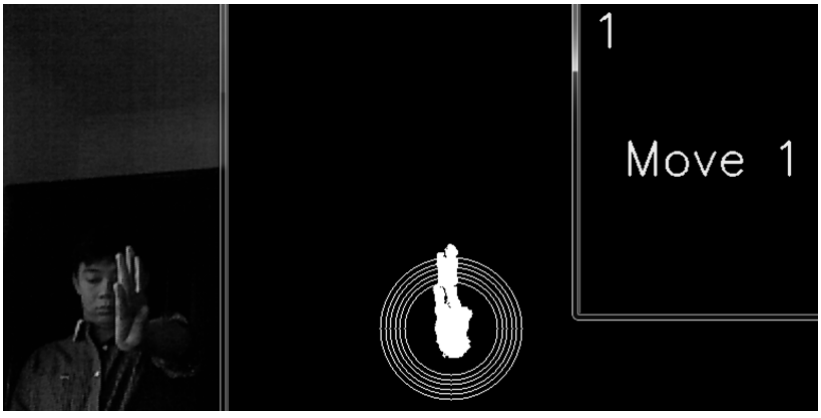


图 7.28　手掌倾斜 80° 后手势识别错误

当手掌倾斜角度过大时，本系统并不能正确识别出手势。实验发现，存在一个临界角，当倾斜角度小于临界角时，识别率正常；当倾斜角大于临界角时，识别率大幅下降。临界角大约为 60° 。主要的原因是：当倾斜角度大于临界角时，在归一化的前景图像中，多个手指会重叠在一起，使识别结果出错。具体的测试数据如表 7.4 所示。

表 7.4 不同倾斜角下手势 4 的识别结果

倾斜角度	测试次数	成功识别次数	手势识别率
30°	200	195	97.5%
45°	200	196	98.0%
60°	200	23	11.5%
80°	200	0	0%

7.2 智能小车的设计与实现

本系统将识别出的手势转化为相应的控制信息[4]，然后通过无线模块将控制指令发送给智能小车，智能小车完成相应的动作。本章将介绍使用的硬件及实现方法。

7.2.1 模块介绍

1. 控制模块

小车的控制模块负责处理与 PC 端的通信，以及控制电机的转动。本系统采用 C51 单片机最小开发板当做控制模块，实物如图 7.29 所示。

控制模块的核心是 STC89C52RC 型单片机。STC89C52RC 单片机是 Intel 公司 MCS-51 系列单片机中最基本的产品。它是采用了 Atmel 公司可靠的 CMOS 工艺制造的 8 位高性能单片机，属于标准的 MCS-51 的 HCMOS 产品。89C52 结合了 CMOS 的高速和高密度技术以及 CMOS 的低功耗特征。它使用标准的 MCS-51 单片机体系结构和指令系统，是 89C51 型单片机的增强版本。89C52 单片机集成了时钟输出和计数器等更多的功能，适用于电机控制等应用。89C52 内置 8 位中央处理单元、256B 片内数据存储器（RAM）、8KB 片内程序存储器（ROM）、32 个双向输入/输出（I/O）口、3 个 16 位定时/计数器、5 个两级中断结构，一个全双工串行通信口和片内时钟振荡电路。此外，89C52 还可工作于低功耗模式，可通过软件选择空闲和掉电工作模式。在空闲模式下，冻结 CPU 但 RAM 定时器、串行口和中断系统维持工作。掉电模式下，89C52 单片机保存 RAM

数据，停止时钟振荡，同时停止芯片内其他功能。89C52 有 PDIP（40pin）和 PLCC（44pin）两种封装形式。

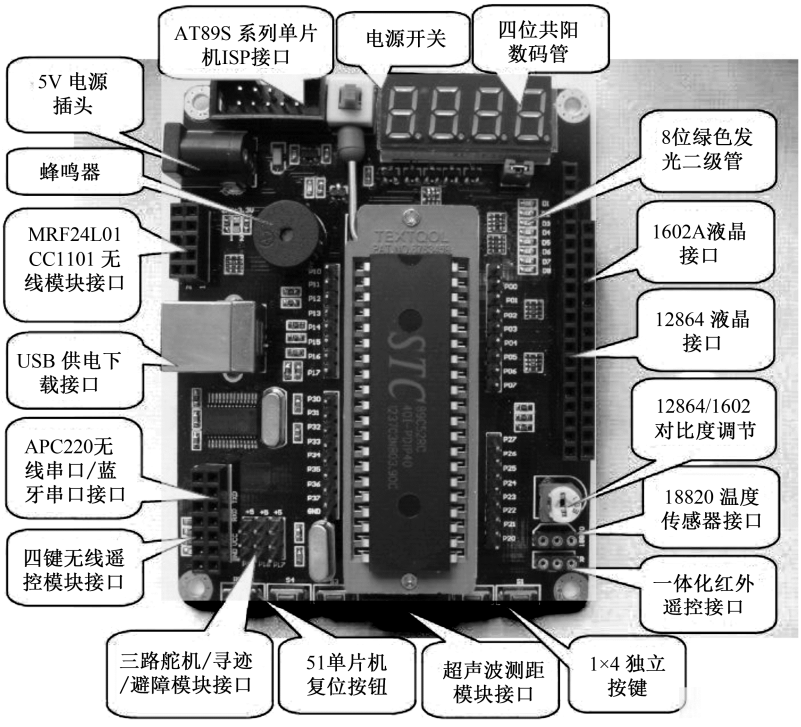


图 7.29　控制模块

2．无线模块

无线模块负责小车与 PC 端的通信。本系统采用的是 APC220 多通道微功率嵌入式无线数据传输模块，实物如图 7.30 所示。APC220 模块是半双工高度集成微功率无线数据传输模块。它嵌入了高速单片机和高性能射频芯片。它创新地采用了高效的循环交织纠检错编码技术。因此，它的抗干扰和灵敏度都大大提高。APC220 模块最大可以纠 24 位的连续突发错误，该点达到了业内领先水平。同时 APC220 模块提供了多个频道的选择，可方便地在线修改发射功率、串口速率、射频速率等各种参数。APC220 模块能够透明传输任何大小的数据，同时体积小、运行电压宽、传输距离较远，并且具有便捷的软件编程设置功能，使之能够有非常广泛的应用。

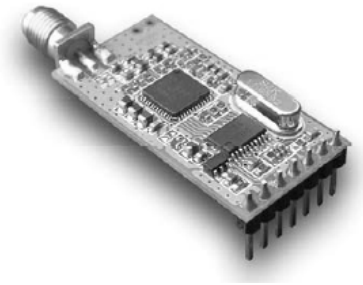


图 7.30　APC220 无线通信模块

APC220 特点：

- 最大传输距离 1000m（2400bps）。
- 工作频率 418M～455MHz（1kHz 步进）。
- 使用 GFSK 调制方式。
- 高效的循环交织纠错编码。
- UART 接口，RS232/RS485 可定制。
- 超大的 256B 数据缓冲区。
- 内置看门狗功能，可保证长期可靠运行。

APC220 模块的使用相当灵活，可以根据用户的需求设置不同的选项。可选的设置参数如表 7.5 所示。

表 7.5　APC220 模块可选设置参数

设　置	选　相	默　认
串口速率（Series Rate）	1200、2400、4800、9600、19200、38400、57600bps	9600bps
串口效验（Series Parity）	Disable, Even Parity（偶效验），0dd Parity（奇效验）	Disable
收发频率（RF Frequency）	418M～455MHz（1kHz 步进）	434MHz
空中速率（Series Rate）	2400、4800、9600、19200bps	9600bps
输出功率（RF Power）	0～9（9 为 20mW）	9（2mW）

第一次使用之前，需将一对或多个 APC220 模块进行参数设置。参数设置十分方便，只需使用自带的 RF-Magic 设置软件就可以完成设置，如图 7.31 所示。

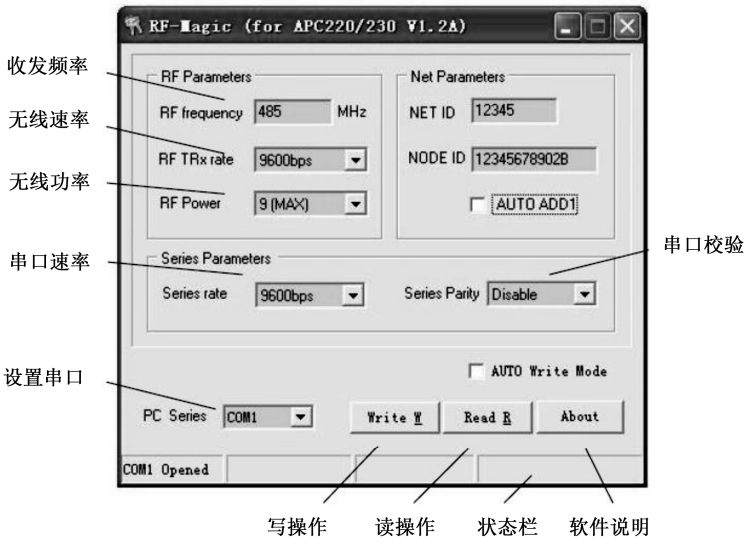


图 7.31　RF-Magic 设置软件

3．电机驱动模块

因为单片机的 GPIO 负载能力不强，不能够直接驱动电机，所以引入了电机驱动模块来驱动电机。本系统使用的是 L298N 电机驱动模块，实物如图 7.32 所示。

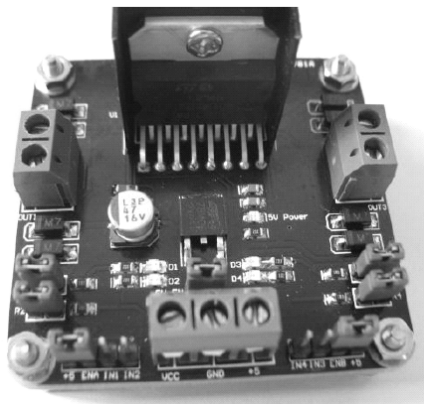


图 7.32　L298N 电机驱动模块

恒压恒流桥式 2A 驱动芯片 L298N 是 SGS 公司推出的产品，比较常见的是 15 脚 Multiwatt 封装的 L298N。它内部包含 4 通道逻辑驱动电路，可以驱动两个

直流电机，或一个两相步进电机。L298N 输出电压最高可达 50V，可以直接通过电源来调节输出电压；可以直接用单片机的 I/O 口提供信号；而且电路简单，使用比较方便。其引脚如图 7.33 所示。

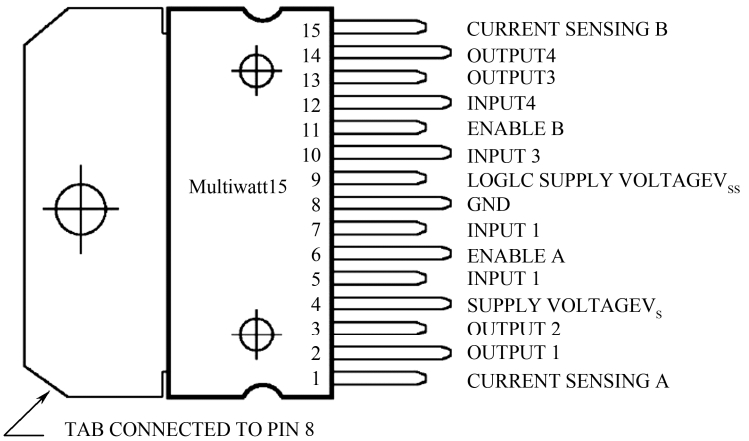


图 7.33　L298N 芯片引脚图

7.2.2　PC 端控制程序

PC 端控制程序完成了三个功能：进行手势识别[5]；将识别出的手势转化为相应的控制信息并在图形化界面上显示当前的手势识别结果与控制信息；通过串口将指令传输到 PC 端的无线模块。界面如图 7.34 所示。手势识别功能已经在第 3 章中介绍，本节就不再赘述。

图 7.34　PC 端控制程序界面

本系统共有两类控制信息：自动运动和手动运动[5]。自动运动是让小车完成预先设定的运动步骤，如“向前移动一米，原地掉头，再移动一米回到初始位置”。共有 4 套预设的自动运动动作，分别对应预设的“1”、“2”、“3”、“4”这四种手势。手动运动是让使用者实时通过手势控制小车的前后左右移动。当用户做出手势“5”时，将手掌的位置转化为相应的小车移动方向：手掌位置偏上，则小车向前移动；偏下则向后移动；偏左则左转；偏右则右转；位置为中间则停车。PC 端的图形化界面通过 OpenCV 的 HighGUI 来实现，如图 7.35 和图 7.36 所示。

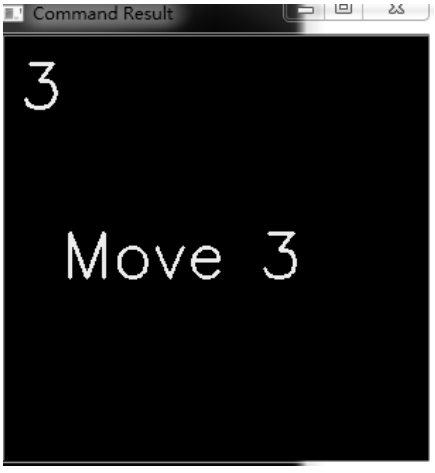


图 7.35　手势“3”对应的自动运动界面

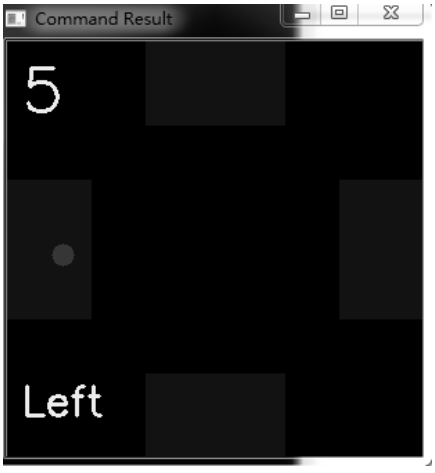


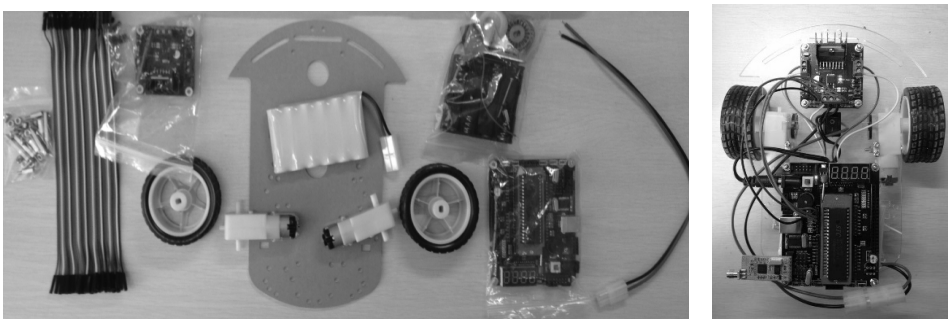
图 7.36　手势“5”对应的手动运动界面

PC 端程序中的 SerialPort 类完成了串口读写控制命令的功能。PC 通过串口与一个 APC220 模块相连。程序每次运行都需要初始化串口：打开串口，设置参数（波特率设为 9600bps、数据位为 8 位、停止位为 1 位），创建串口监听线程来实时监听接收的串口数据。发送控制指令时，只需要直接向串口写入数据即可。为了保证控制的可靠性，PC 与小车采用双向“握手”式通信。PC 通过无线模块发送一条指令之后，小车会在接收到指令并且执行完毕后回传一条确认指令[6]。只有当 PC 收到当前指令的确认指令之后，才会传输下一条控制指令。

7.2.3　智能小车制作与控制

小车的组装步骤：①将各部件使用螺丝固定在车架上；②用导线将各个部件相连。连线方法：APC220 模块直接插在 C51 开发板的串口插槽上；单片机的

P1.1～P1.4 口连接 L298N 的 IN1～4 口；L298N 的 OUT1、2 连接一个直流电机的两端，OUT3、4 连接另一个直流电机的两端；电池先与开关相连接，再连接到 L298N 模块上的 7805 稳压管输入端，稳压管的输出端连接到 C51 开发板的电源接口，如图 7.37 所示。



（a）　　（b）

图 7.37　小车原件和组装后的小车

小车的控制是通过 C51 单片机编程实现的，其程序流程图如图 7.38 所示。

程序开始运行后，首先设定定时器 1，使之产生 9600bps 的波特率。然后打开串口，设定为串行工作模式，并禁用中断。然后程序进入主循环，循环检测串口是否收到了 APC220 发来的数据。如果接收到了指令，则控制小车完成相应的动作。完成动作之后，通过串口向 APC220 写入动作完成指令。最基本的控制指令有五种，前后左右停，指令编号为 1～5。前进动作是由两个驱动轮都向前旋转实现的。此时，控制电机正相输入端的 P1.1 和 P1.3 为高电平，控制反相输入端的 P1.2 和 P1.4 为低电平。同样，后退时两个轮子都向后旋转，P1.2 和 P1.4 为高电平，P1.1 和 P1.3 为低电平。转弯时，一个轮子向前旋转，另一个轮子向后旋转。除了五种基本运动动作以外，还有四种特殊动作。特殊动作的指令编号为 6～9，它是由多个基本动

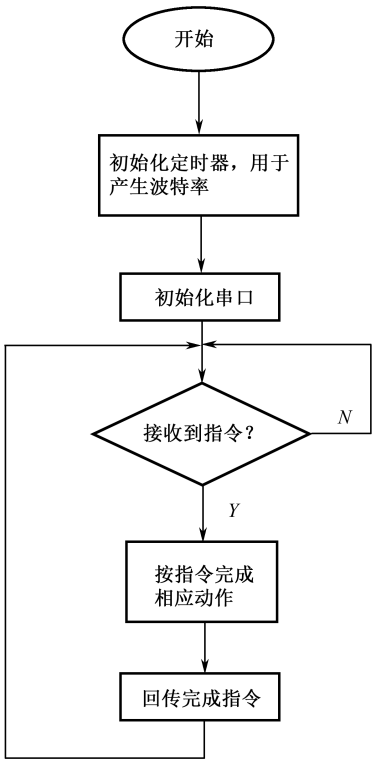


图 7.38　小车控制程序程序流程图

作组合而成的，如特殊动作 1：小车前进 3000ms，左转 1500 ms，停止 4000 ms，右转 2500 ms，后退 4000 ms。

7.3　本章小结

该系统使用 Kinect 获取场景深度图像，然后使用基于深度直方图的自动阈值技术进行手部区域分割，最后使用一个简单而又高效的手势特征分类器进行手势识别。该分类器首先使用数学形态学的图像腐蚀算法找到手掌中心，然后使用“画圈数手指”的方法确定手势的手指数目（特征提取），最后根据特征的不同进行分类。识别出手势之后，将手势转化为相应的控制指令，通过无线模块发送给智能小车。智能小车按照收到的指令来进行相应的移动。

本章参考文献

[1] 陈子豪. 基于深度信息的手势检测与跟踪[D]. 广东：华南理工大学, 2012.

[2] 张毅, 张烁. 基于 Kinect 深度图像信息的手势轨迹识别及应用[J]. 2012， 29（9）：3547-3550.

[3] Jesus Suarez and Robin R. Murphy. “Hand Gesture Recognition with Depth Images: A Review”[C] //The 21st IEEE International Symposium on Robot and Human Interactive Communication.

[4] 吴国斌, 李斌. Kinect 人机交互开发实践[M]. 北京：人民邮电出版社, 2012.

[5] 杨景旭. 利用 Kinect 估计人体头部姿态[D]. 江苏：南京理工大学, 2012.

[6] K. K. Biswas and S. K. Basu, “Gesture recognition using Microsoft Kinect”[J]. Automation, Robotics and Applications (ICARA), pp. 100-103, 2011.

反侵权盗版声明